Computational Models of Perceptual Organization

Stella X. Yu

CMU-RI-TR-03-14

Dissertation submitted in partial fulfillment of the requirements
for the Degree of Doctor of Philosophy in Robotics at
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213, USA

Abstract

Perceptual organization refers to the process of organizing sensory input into coherent and interpretable perceptual structures. This process is challenging due to the chicken-and-egg nature between the various sub-processes such as image segmentation, figure-ground segregation and object recognition. Low-level processing requires the guidance of high-level knowledge to overcome noise; while high-level processing relies on low-level processes to reduce the computational complexity. Neither process can be sufficient on its own. Consequently, any system that carries out these processes in a sequence is bound to be brittle. An alternative system is one in which all processes interact with each other simultaneously.

In this thesis, we develop a set of simple yet realistic interactive processing models for perceptual organization. We model the processing in the framework of spectral graph theory, with a criterion encoding the overall goodness of perceptual organization. We derive fast solutions for near-global optima of the criterion, and demonstrate the efficacy of the models on segmenting a wide range of real images.

Through these models, we are able to capture a variety of perceptual phenomena: a unified treatment of various grouping, figure-ground and depth cues to produce popout, region segmentation and depth segregation in one step; and a unified framework for integrating bottom-up and top-down information to produce an object segmentation from spatial and object attention.

We achieve these goals by empowering current spectral graph methods with a principled solution for multiclass spectral graph partitioning; expanded repertoire of grouping cues to include similarity, dissimilarity and ordering relationships; a theory for integrating sparse grouping cues; and a model for representing and integrating higher-order relationships. These computational tools are also useful more generally in other domains where data need to be organized effectively.

Keywords: perceptual organization, image segmentation, figure-ground, depth segregation, attention, bias, popout, visual search, clustering, graph partitioning, constrained optimization.

To my parents and brother.

Acknowledgments

Looking back at my years as a graduate student at Carnegie Mellon, I consider myself most fortunate to have met the following mentors and friends.

My thanks first go to Jianbo Shi, for many exciting insights he shared with me on various topics of research. It is Jianbo who brought me to the frontier of computer vision research. It is also Jianbo who stimulated my independence. Finding faith in myself was the biggest turning point in my career, and I forever owe this to Jianbo.

To Tai Sing Lee, who has poured a lot of attention and effort in developing a neuroscientist out of an engineer. His endless patience, inviting conversations and invaluable guidance has cultivated my thinking on vision science. I will also cherish his friendship in the years to come.

To Takeo Kanade, who has enthusiastically supported my research in neuroscience as well as computer vision. His diligence, wisdom, humor, and clear thinking on research problems never cease to inspire me.

To Ted Adelson, Martial Hebert and Mike Lewicki for serving on my thesis committee, for broadening my view on my work and giving me constructive feedback in a most friendly and kind manner.

Special thanks go to Shyjan Mahamud for many exciting discussions on research problems and sharing all his part detection results, to Jing Xiao for providing me with his head-tracking image sequences, to David Tolliver and Vandi Verma for excellent comments on my thesis draft and talk slides, and to Sonya Allin for proofreading the final version of the thesis.

To all the members in the CMU Microdynamics Lab, the CMU HumanID Lab and the Center for the Neural Basis of Cognition, in particular, Ralph Hollis, Arthur Quaid, Jay Gowdy, Al Rizzi, Peter Metes, Xuhui Zhou, Jeff Cohn, Bob Collins, Yanxi Liu, Fernando De la Torre, Ralph Gross, Iain Matthews, Hua Zhong, Leon Gu, Yan Li, Jake Sprouse, Jay McCleland, Carl Olson, Carol Colby, Rick Romero, Xiaogang Yan, Vivek Khatri, Roman Mitz for embracing my bluntness with such warm tolerance, and for their perpetually enthusiastic help and encouragement.

To Julie Rollenhagen, Stewart John Moorehead and Shyjan Mahamud for being the best of friends, from whom I learn so much about life, with whom my joy and happiness multiply, and with whom my worry and sorrow dissipate.

To Su Yin, Yueming Yu, Deyun Chen, Mei Han, Wei Hua, Dongmei Zhang, German Cheung, Jing Xiao, Bob Wang, Wei Tech Ang, Jonas August, Tsuyoshi Moriyama, Chuck Rosenberg, Pinar Duygulu, Qifa Ke, Yanghai Tsin, Jinxiang Chai and fellow students at the Robotics Institute, for making this place so interesting and this long journey so enjoyable.

To staff members Anita Connelly, Darlene Kapcin, Jackie Jenkins, Monica Hopes, Louise Ditmore, Stephanie Matvey and particularly our Robo Moms Marce Zaragoza and Suzanne Lyons Muth, for fixing loose ends now and then and making my life so much easier!

I would also like to thank the School of Computer Science at CMU, for providing its trainees with an unparalleled learning environment. This research was sponsored by DARPA HumanID ONR N00014-00-1-0915, NSF IRI-9817496, NSF LIS 9720350 and NSF 9984706. Real images used in this thesis are publically available from fashionshowroom.com, berkeley.edu, and nytimes.com.

Finally, I express my love and gratitude to my family, for their absolute confidence in me and for their understanding and support of my pursuits. To Yu Liu, the man who has come into my life and has always been there for me since. I dedicate this thesis to my parents, who bore me, raised me and infused me with boundless curiosity and persistence; and to my brother, who leads me, protects me and offers me his best from the day I came into this world. Thanks!

Contents

Chapter 1

Introduction

The ability of the human visual system to isolate *objects* from an image is remarkable, yet the computational underpinnings of such an ability remain elusive.

Consider a few examples of images taken from our everyday lives. Fig 1.1 is a sample of simple scenarios, where the foreground and background can each be described by a set of features, for example, image intensity. A qualitative description of how we perceive the structures exhibited in such visual inputs was first put forward by experimental psychologists in the 19th and 20th century, who termed the process *perceptual organization*. A set of *laws of grouping* under ideal simplistic settings for artificial stimuli were derived: all else being equal, elements are structured into groups sharing a common feature, e.g. intensity, color, or motion (Wertheimer, 1938; Kanizsa, 1979; Palmer, 1999). Consequently, earlier attempts at image segmentation focused on simple low-level feature analysis and enhancement, giving rise to approaches such as thresholding (Sahoo et al., 1988) and edge linking (Witkin, 1983; Canny, 1986; Deriche, 1990).

In a basic thresholding approach, a histogram of intensities for all the pixels in the image is first collected. Then a threshold is found by locating the deepest valley of the histogram to maximize the intensity difference between two pixel sets. This approach only looks at the statistics of intensity values without taking their spatial distribution into account. When it is applied to the first image in Fig 1.1, for example, parts of the face and hands become the background. To

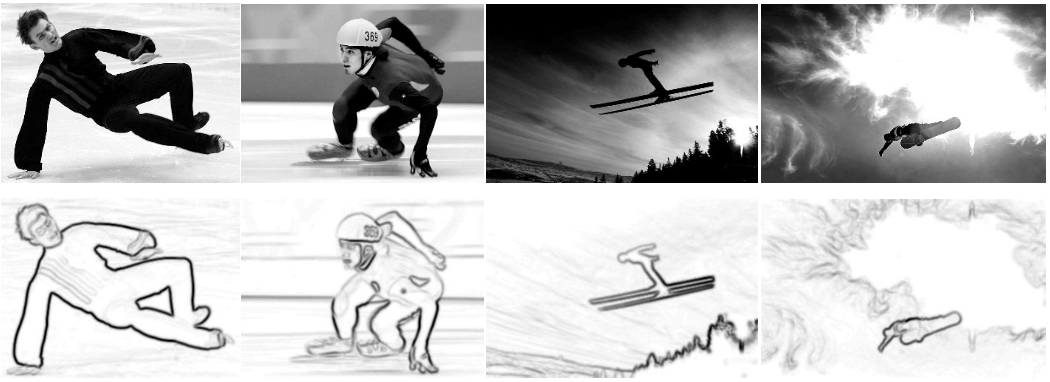


Figure 1.1: Images of simple foreground and background can be segmented based on feature attributes, such as brightness and spatial coherence. Row #1: images. Row #2: edge magnitudes obtained at a fixed scale. Same convention for Fig 1.2 to Fig 1.4.

make up for the ignored spatial aspect of segmentation in thresholding approaches, morphological operations (Jain, 1989) are often used subsequently, which poses a tradeoff between removing noise and preserving small structures.

Compared to thresholding techniques, edge linking is a spatially local approach, where the division between two regions is found by detecting changes in low-level features within a local neighbourhood. As can be seen in the second row of Fig 1.1, edge detection is more robust to global transient changes in lighting. However, it is non-trivial to derive a region segmentation from detected edges since edges are not guaranteed to form closed contours. This is a common problem with most contour-based approaches. Another common problem with detected edges is that they may not correspond to the boundaries of objects. Texture of an object can lead to strong firing of an edge detector, while true object boundaries may have weak contrast against the background.

In Fig 1.2, the foreground object is still clearly delineated from the background, even though the images are considerably noisier than in the previous figure. Now, segmentation based on just a common feature value for each region is insufficient. To handle such situations, further grouping laws such as boundary smoothness need to be employed. Knowledge of such generic priors on boundary and/or region properties is used in various approaches of image segmentation:

active contours (Kass et al., 1988; Cohen, 1991; Ronfard, 1994), boundary detection (Mumford and Shah, 1985; Nitzberg et al., 1993; Geiger and Kumaran, 1996; Williams and Jacobs, 1997), region growing (Hong and Rosenfeld, 1984; Adams and Bischof, 1994), region competition (Zhu and Yuille, 1996), Markov random fields (Geman and Geman, 1984; Blake and Zisserman, 1987), level-set and variational methods (Sethian, 1996), graph cuts (Shi and Malik, 2000), etc.

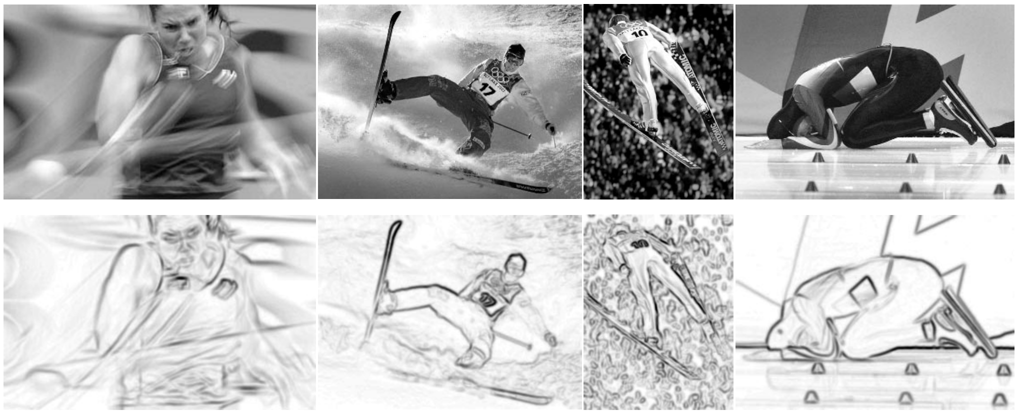


Figure 1.2: Poor imaging conditions and the interactions between foreground and background can confuse a segmentation algorithm based solely on feature similarity. Regions are no longer characterized by their feature values alone, and their boundaries do not always have sharp contrast. Boundary smoothness, symmetry, focus and defocus, all become important cues in sorting out competing segmentations.

Examples of such generic priors are the piecewise homogeneity assumption and the smooth boundary assumption. In its basic form, the former assumes that an image region has roughly the same intensity, while neighbouring regions have different intensities. The latter assumes that object boundaries are smooth, while those from surface textures or spatial displacement between objects tend to have abrupt turns and cusps. When we look for regions or boundaries with desired properties, our image segmentation is driven away from those presumably distracting image features. Various other assumptions have also been explored, e.g., convexity (Jacobs, 1996) and closure (Mahamud et al., 2003) of boundaries.

Fig 1.3 presents challenges other than noise: occlusion and clutter. Segmentation for such images cannot depend merely on low-level signal processing. Specific knowledge about objects and their relationships to the environment has to be incorporated into the segmentation process. Examples of such segmentation schemes include model-based inference methods: Hough transforms (Illingworth and Kittler, 1988), geometric hashing (Wolfson and Rigoutsos, 1997) and deformable templates (McInerney and Terzopoulos, 1996). The distinction between segmentation and recognition becomes vague.

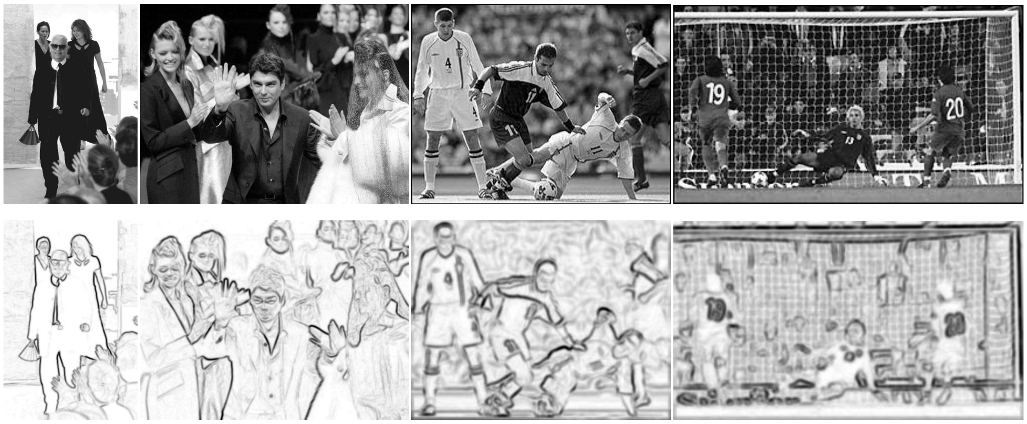
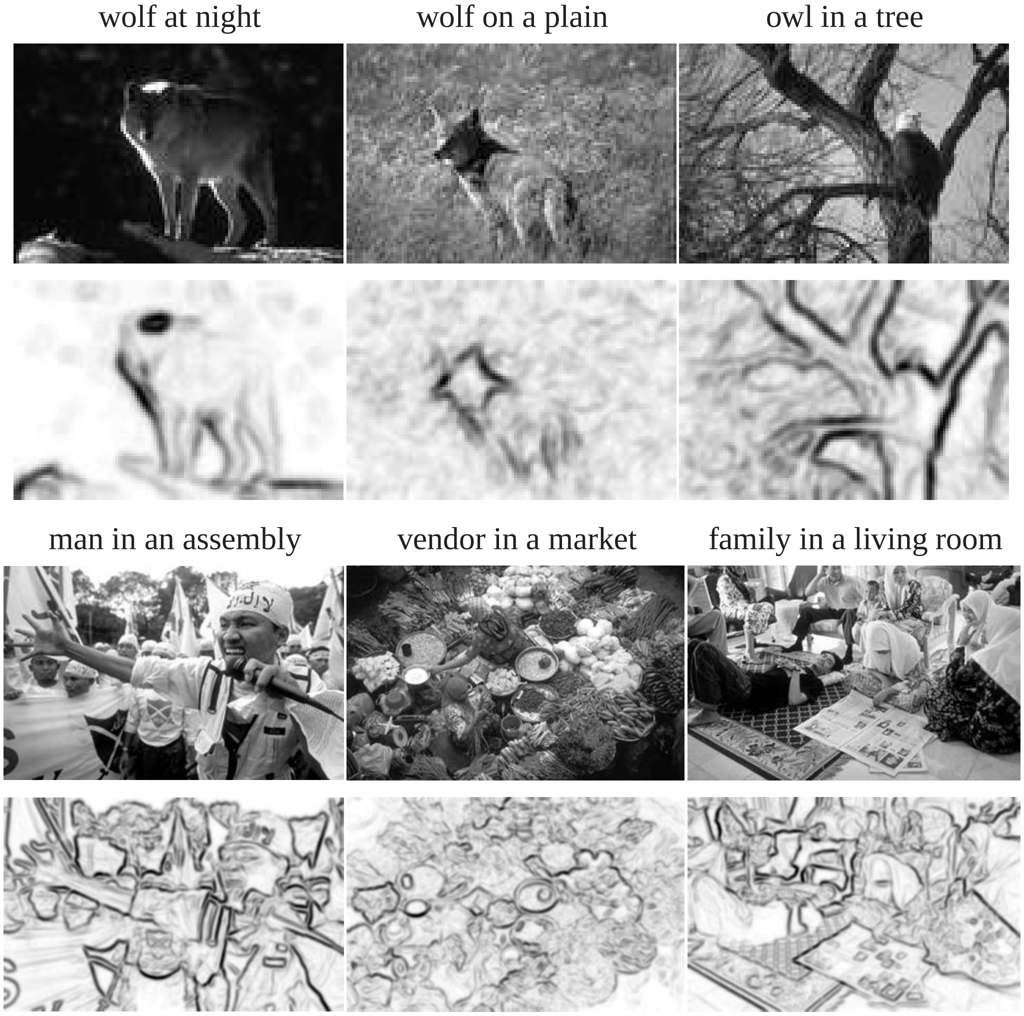


Figure 1.3: Clutter and occlusion pose another major challenge in image segmentation. As a result, significant portions of region boundaries can vanish. Likewise, in highly textured regions, focused visual processing is required for a human subject to delineate the part belonging to a foreground object. Object segmentation is impossible without prior knowledge about object shapes and articulated configurations.

For example, when using Hough transforms to detect circles in an image, we first parameterize a circle with its location and size. Image features such as corners or edges are detected and votes for all possible realizations of a circle are tallied. The one with the most votes becomes the recognized circle. The problem is that low-level measurements are often noisy, thus the votes are not reliable. In addition, if the object model involves many parameters, detecting the winning realization is problematic, since there may not be enough votes from image features. Deformable templates constrain the global object shape in a more rigorous way, however, they often critically depend on initialization.

In Fig 1.4, it becomes evident that recovering objects that are barely distinguishable from clutter is difficult without an understanding of the scene context (Yarbus, 1967; Rimey, 1993). Perceptual processing under such adverse conditions is what makes computer vision one of the most challenging fields of artificial intelligence.



wolf at night

wolf on a plain

owl in a tree

man in an assembly

vendor in a market

family in a living room

Figure 1.4: For images of outdoor scenes, objects of interest are often indistinguishable from their surroundings. When both foreground and background are richly textured, or when the scene is taken under rare viewing directions, it is extremely difficult to segment objects of interest without attempting an interpretation of the rest of the scene.

Finally, the computational modeling of visual perceptual organization is at its roots the study on how to organize information effectively. Other domains such as sensory and text processing also share the same underlying principles of organization. The study of such principles, however, are most readily intuitive in vision. Perceptual processing of images has all the complexity that is present in the other domains, yet the processing can be easily scrutinized and comprehended. This thesis focuses specifically on the computational modeling of perceptual organization, however the tools developed can be easily adapted to other data grouping problems as well.

1.1 Overall Approach

Theories for perceptual organization roughly fall into two camps: the process is either sequential or interactive. See Fig 1.5.

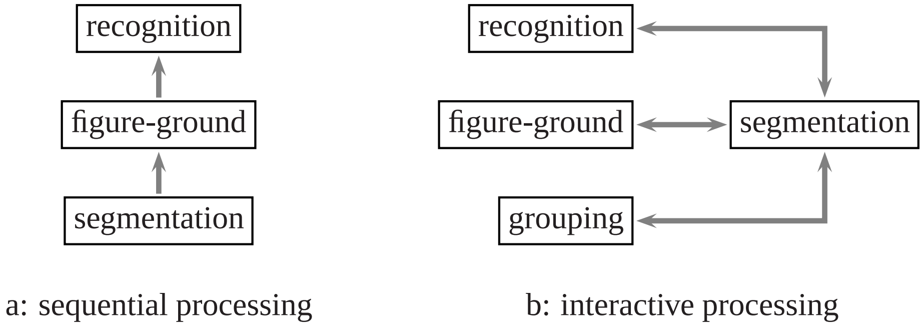
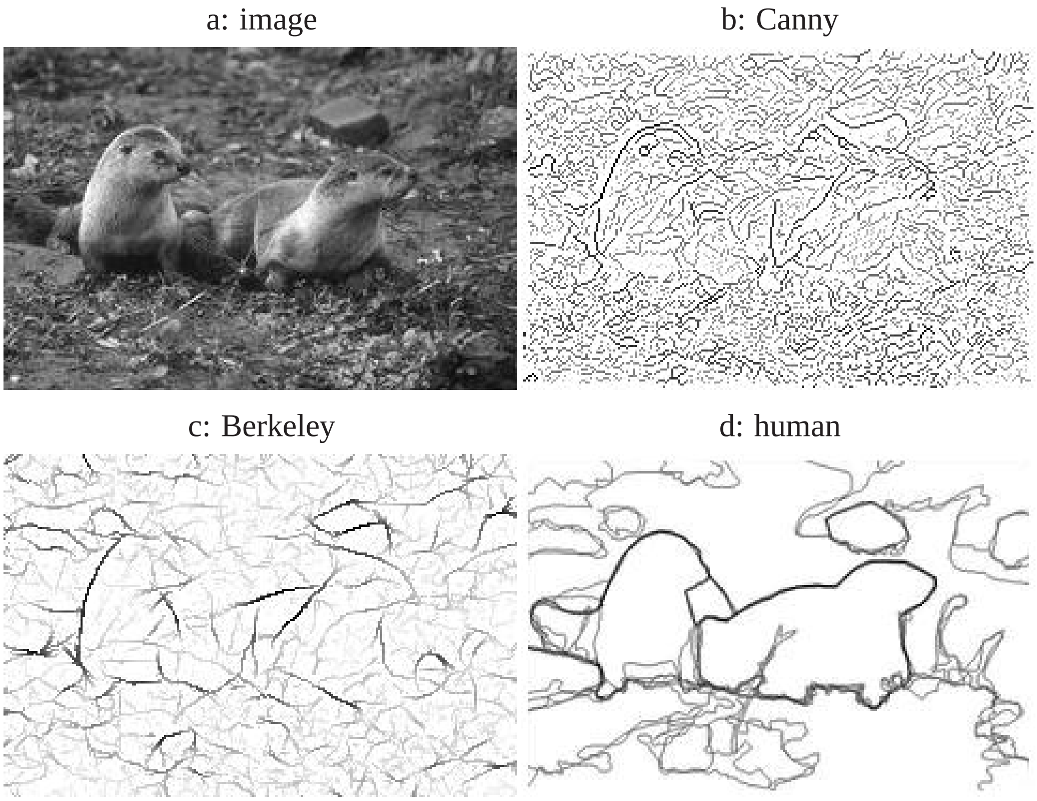


Figure 1.5: Two views on perceptual organization. a: Segmentation is a grouping process acting on low-level cues, e.g. intensity, edgels etc. The resulting regions are further grouped into foreground and background based on higher-level cues, e.g. convexity, symmetry, parallelism etc. Only the foreground is processed further for object recognition. b: Segmentation is considered the outcome of an interactive process among high-level object knowledge, intermediate figure-ground cues and low-level grouping cues.

In the sequential processing theory popularized by Marr (Marr, 1982), visual processing starts with what is possible to compute directly from an image and ends with the information required to support goals such as navigation or object recognition. In-between representations are derived to turn the available information at one level to the required information at the succeeding level. Accordingly, most current image segmentation algorithms adopt a bottom-up approach. They start with an over-segmentation based on low-level cues such as feature similarity and boundary continuity, and then build up larger perceptual units (e.g., surface, foreground and background) by adding high-level knowledge (e.g., statistical properties of regions) into the grouping process (Zhu and Yuille, 1996). Identifying perceptual groups by a generic segmentation process helps in discovering the underlying causes of perceptual phenomena (Hoffman, 1983; Witkin and Tenenbaum, 1983; Freeman, 1996; Knill and Richards, 1996), achieving perceptual constancy (Adelson, 1999; Adelson and Pentland, 1996; Hochberg and Brooks, 1962; Lowe, 1984), or compressing the redundant representation by coding only relevant information (Attneave, 1954; Barlow, 1960; Mumford, 1996).

Although a sequential system can relieve later stages of perceptual processing of computational burden, such a feed-forward system is vulnerable to mistakes made at each step. The reason why it is vulnerable is that it always faces a chicken-and-egg dilemma. Without utilizing any knowledge about the scene, image segmentation gets lost in poor data conditions: weak edges, shadows, occlusions and noise (Fig 1.6). Missed object boundaries are often hard to recover in subsequent processing. Gestaltists have long recognized this issue, circumventing it by adding a grouping factor called *familiarity* (Palmer, 1999). On the other hand, without being subject to perceptual constraints imposed by low-level grouping, an object detection process can produce many false positives in a cluttered scene (Kanizsa, 1979; Jacobs, 1992; Mahamud, 2002). Eliminating such false positives by checking image data directly against object models not only is time-consuming, but also frequently ends up hallucinating objects (Fig 1.7). Locally, many of these hallucinations have features resembling some known objects, but in a larger context surrouding them, they are not distinctive enough to validate the known objects. Such contextual analysis is what perceptual organization does and that’s where perceptual organization can help object recognition.



a: image

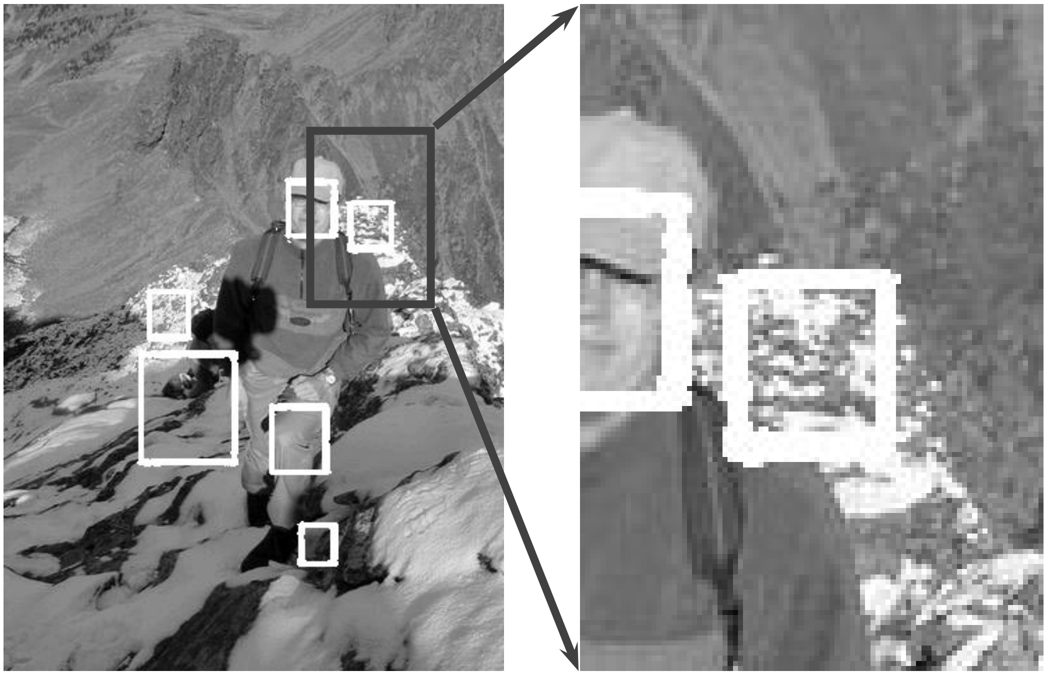
b: Canny

c: Berkeley

d: human

http://www.cs.berkeley.edu/projects/vision/grouping/segbench/pb/index.html

Figure 1.6: Low-level image segmentation alone often does not respect object boundaries. a: Shown here is an image example for edge detection by (Martin et al., 2002). b: edges by Canny edge detector, the goal of which is to detect sharp changes in pixel intensities. It fires strongly in textured areas (e.g. the stony ground), although such intensity changes do not correspond to object boundaries. c: edges by (Martin et al., 2002), where human segmentation data were used to learn a boundary classifier based on local brightness, color and texture cues. Many sporadic texture edges are suppressed. d: segmentation by human subjects, where object boundaries are clearly marked despite the lack of intensity contrast at some places. Significant improvements have been made in predicting object boundaries directly from low-level feature statistics. However, without the guidance of global object-level knowledge, low-level image segmentation still easily misses object boundaries due to lighting and clutter.



http://vasc.ri.cmu.edu/demos/faceindex

Figure 1.7: Object detection schemes are often overwhelmed by false alarms. Shown here is a face detection example by (Schneiderman and Kanade, 2002), where a set of low-level features are first learned from training images, and then used directly to classify test image patches as faces/non-faces. This detector achieved the state-of-the-art benchmark performance in computer vision, yet false alarms are still inevitable for such images obtained from public submissions. Locally, each false positive has certain features resembling a face, but they are explained away by the surrounding context. In general, an object recognition scheme that does not rely on any image segmentation is prone to false alarms.

In contrast, the interactive processing point of view acknowledges the chicken-and-egg nature of perceptual organization (Kelly and Grossberg, 2000; Rumelhart and McClelland, 1986; Grenander and Miller, 1994), and overcomes the issue by engaging all perceptual processing simultaneously. The complicating issue that arises is the interactions among the various perceptual modules. Despite agreement on the need for interactive processing in the literature (Peterson, 1994), consensus on the details that realize such a scheme has not yet been reached.

The FACADE theory given in (Kelly and Grossberg, 2000) is a comprehensive computational model constrained by the findings in biological vision systems yet able to explain many peceptual pheonomena. However, it is overly flexible and lacks a clear formulation of the overall computation. Some theories focus on interpreting the computation carried out in biological systems (Lee and Mumford, 2003) with high-level concepts only. Parallel distributed processing theory (Rumelhart and McClelland, 1986) is again a biologically inspired computational framework for perceptual processing. It has revolutionized many concepts regarding the representation and interactions between processing modules. However, such ideas have mainly been demonstrated on artificial stimuli, thus their application is often very limited in scope (Vecera and O'Reilly, 1998).

In computer vision, the most influential interactive processing framework is pattern theory (Grenander and Miller, 1994). The key novel idea is that perceptual processing is an analysis process that is equivalent to synthesizing sensory input with known patterns. It has inspired a whole range of generative approaches. One of the most successful applications of this "analysis = synthesis" idea on segmenting real images is given in (Tu and Zhu, 2002). However, despite its rigorous theoretical basis, the implementation often involves too many parameters and heuristics.

In this thesis, we adopt the interactive processing point of view for the above mentioned reasons. Illustrated in Fig 1.8, our computational models always have a clear objective to optimize. This objective quantifies the concept of *Pragnanz* (Koffka, 1935), which gauges the goodness of an overall perceptual organization by taking cues at all levels into account simultaneously. Unlike the sequential processing point of view, in our work, there is no distinction between segmentation, figure-ground and object recognition. They are merely different projections of one underlying perceptual output fulfilling the Pragnanz.

In our work, interactions within and between visual modules are captured in a graph-theoretic framework. For each module, perceptual elements such as pixels, edgels or patches are represented by nodes, and relationships between the elements are represented by weights attached to edges connecting the nodes.

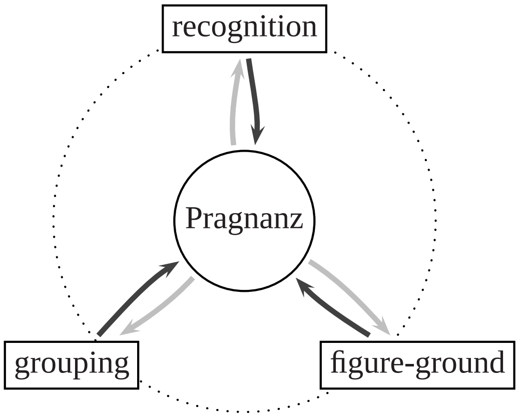


Figure 1.8: Our approach unifies grouping, figure-ground and recognition in one computational framework. The central idea tying the three processes is “Pragnanz”, which gives a goodness measure for a given perceptual organization in terms of all grouping cues, figure-ground cues and object knowledge.

Interactions between modules are represented as constraints on grouping schemes. Finding a good grouping then becomes a node-partitioning problem subject to the interaction constraints.

Previous work (Shi and Malik, 2000) proposed a specific graph partitioning criterion for perceptual grouping. A remarkable property of the criterion is that it has an efficient computational solution: near-global optima of this NP-complete problem can be obtained through the spectral decomposition of a matrix.

Building upon this work on spectral graph theory, this thesis makes the following contributions.

1. We unify grouping cues, figure-ground cues and depth order cues in one process. In particular, we gain more understanding on perceptual popout with respect to general grouping, and we carry out region segmentation and depth segregation at the same time.

2. We unify top-down and bottom-up information in a single grouping process. In particular, our biased grouping process incorporates cues derived from spatial and object attention. The former provides partial grouping cues on visual elements, the latter provides patch grouping cues for specific objects.

The above contributions are achieved by the following novel computational tools that we develop:

1. a principled solution for multi-class spectral graph partitioning;
2. an expanded repertoire of grouping cues, which now include similarity, dissimilarity and ordering relationships.
3. a theory for integrating sparse grouping cues;
4. a model for representing and integrating higher-order relationships.

Summarizing, the salient aspects that distinguish this thesis from prior models of interactive processing are: (1) a criterion formulating the goal of the whole computational process, (2) a fast solution for near-global optima of the criterion, and (3) results on a wide range of real images.

1.2 Road Map

The organization of this thesis is summarized in Fig 1.9. Below, we describe the logical development behind the various chapters.

Chapter 2 gives a principled account of multiclass spectral clustering. Since all models in this thesis are cast in the framework of spectral graph partitioning, this chapter provides a foundation for the rest of the chapters. We first generalize the normalized cuts criterion in graph theory to multiclass problems. A relaxed continuous solution is found by eigen-decomposition. We clarify the role of eigenvectors as a generator of all continuous optima. We then solve an optimal discretization problem, which finds nearly global-optimal discrete solutions closest to the continuous optima. Our method is robust to a random initialization and converges faster than other clustering methods. Flexible initializations also allow us to obtain nearly optimal solutions with special requirements. Extensive experiments on real image segmentation are reported.

Next we enhance our grouping scheme along two lines: by expanding the repertoire of grouping cues, and by guiding grouping with prior knowledge.

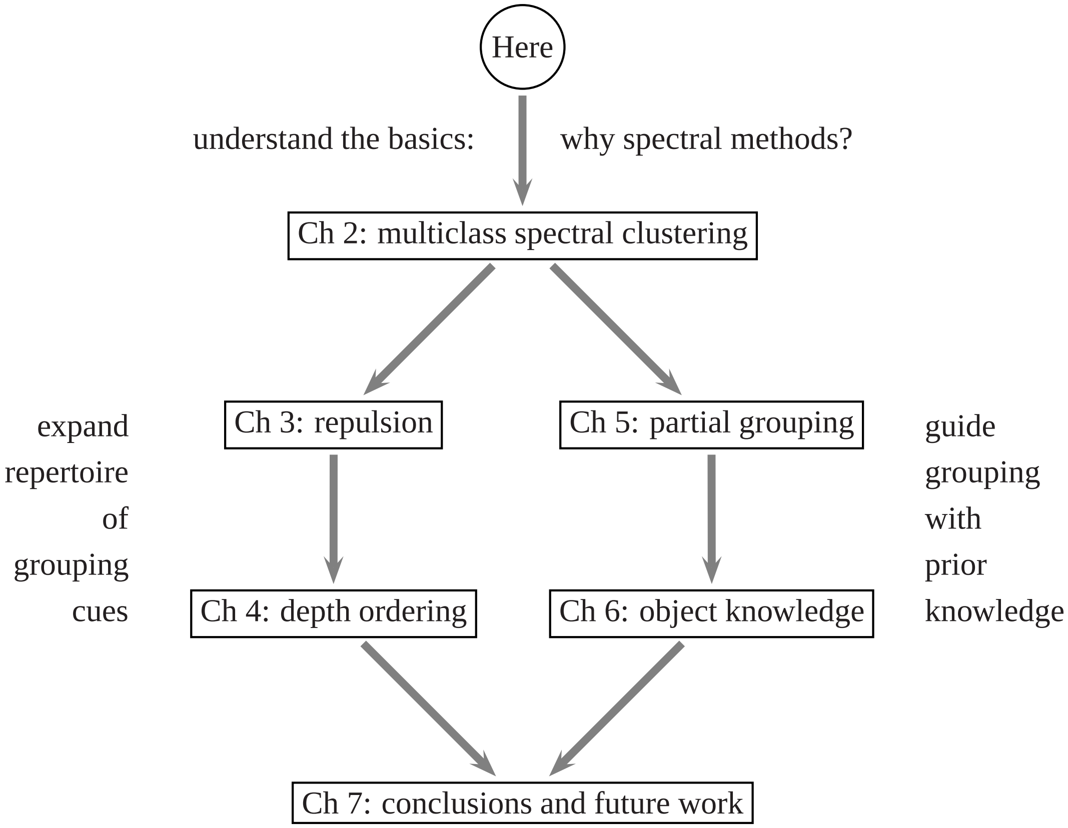

Here
understand the basics:
why spectral methods?
Ch 2: multiclass spectral clustering
expand repertoire of grouping cues
Ch 3: repulsion
Ch 5: partial grouping
guide grouping with prior knowledge
Ch 4: depth ordering
Ch 6: object knowledge
Ch 7: conclusions and future work

Figure 1.9: Road map of this thesis.

In Chapter 3, we identify the active role of dissimilarity in grouping, an often overlooked grouping principle we call *repulsion*, in contrast with *attraction* which groups by similarity. Using attraction for similarity grouping, repulsion for dissimilarity grouping, we provide a theoretical basis for regularizing a solution for spectral graph algorithms. We show that attraction, repulsion and regularization each contributes in a unique way to perceptual grouping.

Further expanding the repertoire of grouping cues, in Chapter 4 we propose the use of ordering cues arising from occlusion events to be handled together with the reciprocal relationships of similarity and dissimilarity cues. This new representation allows for the integration of local grouping and figure-ground cues so that depth segregation may be done simultaneously with region segmentation.

While the methods in Chapter 3 and 4 enrich the representations of graph partitioning approaches, those in Chapter 5 and 6 deal with utilizing prior knowledge to guide the grouping process.

The first form of prior knowledge we consider is partial grouping information known beforehand. We formulate such a biased grouping problem as a constrained optimization problem, where structural properties of the input data define the goodness of a grouping, and partial grouping cues define the feasibility of a grouping. A key observation is that a direct integration of these two sources of information is ineffective. The often sparse partial grouping cues have to be propagated, for which we provide a principled approach. We apply our method to real image segmentation problems, where partial grouping priors can often be derived based on a crude spatial attentional map, i.e. places of salient features, large motion in a video sequence, or prior expectation of object locations. We demonstrate not only that it is possible to integrate both image structures and priors in a single grouping process, but also that objects can be segregated from the background without using specific object knowledge.

The second form of prior knowledge that we consider is object-specific. For example, we want to segment an image into foreground and background, with foreground containing solely objects of interest known *a priori*. Adopting image patches from training sets as a representation for objects, we develop an object segmentation method that incorporates both edge detection and object part detection results. It consists of two parallel processes: low-level pixel grouping and high-level patch grouping. We seek a solution that optimizes a joint grouping criterion in a reduced space enforced by grouping correspondence between pixels and patches. In essence, the output of object patch grouping provides partial grouping constraints on pixels. However, these grouping constraints are not known in advance, since the patch grouping itself is part of the solutions that we seek. With partial pixel-grouping cues dependent on patch grouping, we achieve object recognition and image segmentation at the same time. We report promising experimental results on a database of objects subject to clutter and occlusion.

We conclude the thesis in Chapter 7 with discussions on future work.

Chapter 2

Multiclass Spectral Clustering

Generative models are often used in standard approaches to clustering data points. For example, under a mixture density model, parameter estimation techniques are employed to learn both the model parameters and class labels (Jordan, 1999). Recently a model-based view was also given to common clustering algorithms such as K-means and agglomerative algorithms (Kamvar et al., 2002). With the underlying assumptions explicitly made, the generative approaches are favored for their descriptive properties.

However, two issues limit their success in applications: model inadequacy and/or computational intractability. For example, the popular Gaussian density assumption is rarely appropriate for capturing the complexity of real data; secondly, their energy functions often have many local minima, which are sometimes obtained with very slow convergence.

Consider the point set in Fig 2.1. It has four clusters based on proximity. All four clusters are point clouds, except the first one which has a ring structure. A Gaussian density model well describes a point cloud, but it is poor at capturing the ring cluster. A mixture of Gaussians are used, which introduce a lot more parameters such as the number of Gaussians, the mixing proportion of each component Gaussian in addition to its location, size and shape (Fig 2.2). In high-dimensional spaces, we may not have access to enough training data to constrain the search for optimal parameters, resulting in slow convergence to local optima.

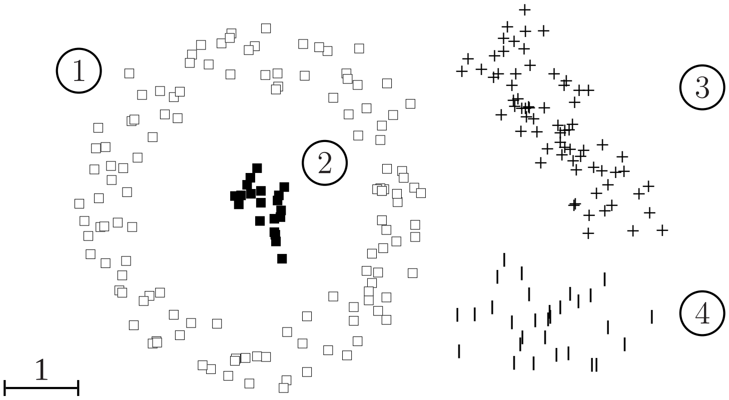


Figure 2.1: Point set data. Points are numbered sequentially in four marked groups, with 125, 20, 70, 30 points each.

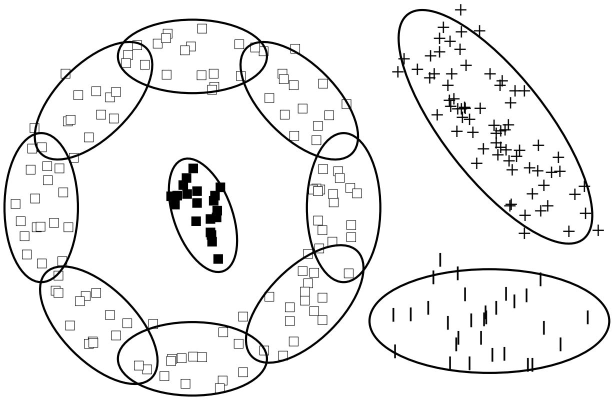


Figure 2.2: Generative approaches for clustering. Shown here is an example of a mixture of Gaussians that explain the point set data. Model complexity can increase tremendously when point sets become more complicated, rendering the computation intractable.

Instead of employing a generative model to explain data clusters, spectral graph partitioning methods (Chung, 1997) are an alternative that is more closely related to multi-dimensional scaling and locally linear embedding (Roweis and Saul, 2000), the goal of which is to preserve certain relationships (e.g. distance) among data points in a lower-dimensional representation. K-class spectral clustering is regarded as one that embeds data points with certain grouping relationships into a K-dimensional space. In contrast with model-based approaches, grouping based on such relational cues can easily adapt to complex data clusters.

In graph-theoretic approaches, each data point is taken as a node. A weighted graph is then built with similarity measures attached to edges connecting the nodes. Clustering points becomes a node partitioning problem. For certain partitioning criteria (Shi and Malik, 2000), the global-optima are obtained in a relaxed continuous domain by solving an associated eigenvalue problem.

To make the idea more concrete, consider Fig 2.1 again. We build a graph with 245 nodes, with a 245×245 matrix W summarizing all possible pairwise relationships between them. In this case, $W(i, j)$ is large if points i and j are close:

$$W(i,j) = \exp\left(-\frac{(\underline{i} - \underline{j})^2}{2\sigma_d^2}\right), \tag{2.1}$$

where $\underline{i}$ denotes the coordinates of data point i. For the point set in Fig 2.1, we set $\sigma_d = 0.5$. Unlike the generative approaches, these cues encode proximity without resorting to any assumption of a global structure. We end up with a graph which tends to have strong connections between nodes within each cluster and weak connections between clusters. A good clustering corresponds to a partitioning scheme that separates all the nodes by cutting off the weakest links among them. This can be formulated as an optimization problem on the weight matrix W. It has been shown in (Shi and Malik, 2000) that for the 2-class normalized cuts criterion, the global optimum in the relaxed continuous domain is given by the second largest eigenvector shown in Fig 2.3. Using the criterion to select the best threshold on this eigenvector, we can divide all points into two classes, which correspond to the left and right division between clusters 1, 2 and 3, 4.

Overall, spectral graph methods are conceptually simple, numerically efficient (Anstreicher and Wolkowicz, 2000), and successful in applications such as circuit layout (Chan et al., 1994; Alpert and Kahng, 1995a), load balancing in parallel computation (Hendrickson and Leland, 1995) and image segmentation in computer vision (Malik et al., 2001).

However, the conceptual simplicity of such approaches is lost in the last step which involves recovering a discrete solution from the continuous solution. A majority of the theoretical work on spectral methods have dealt with bipartitioning

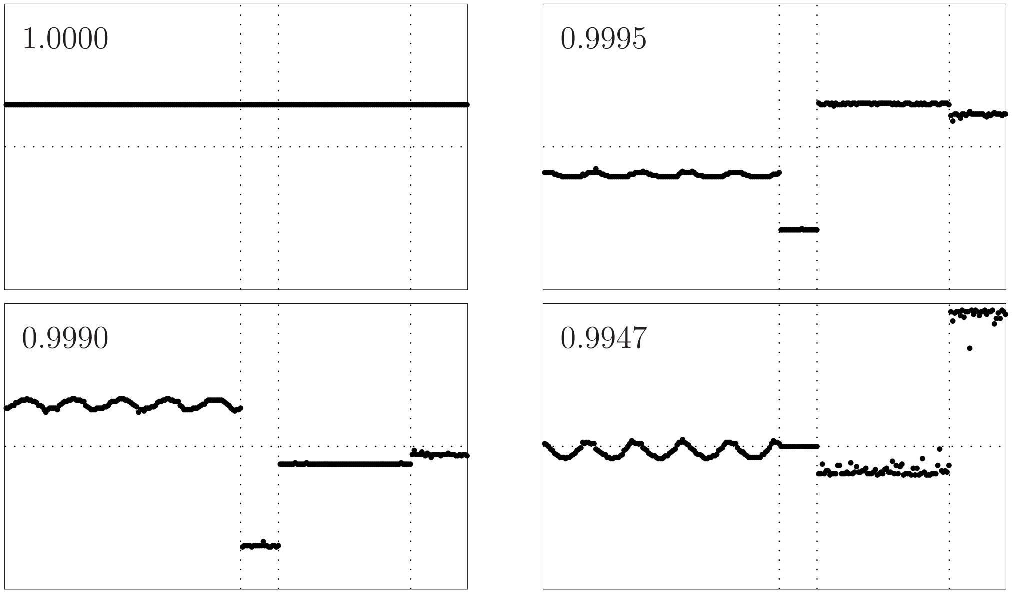


Figure 2.3: The first four eigenvectors of normalized cuts ordered according to their eigenvalues given in each plot. Horizontal dotted lines indicate 0. The vertical lines denote the four partitions.

(Chung, 1997). For K-way partitioning, most previous works treat the eigenvectors as a lower-dimensional geometric embedding (Alpert and Kahng, 1995b) of the original problem, where data points in the same class take similar eigenvector components and are thus mapped to nearby n-dimensional points, with each coordinate specified by one of the n eigenvectors. Various clustering heuristics such as K-means (Shi and Malik, 2000; Ng et al., 2002), transportation (Barnes, 82), dynamic programming (Alpert and Kahng, 1995b), greedy pruning or exhaustive search (Shi and Malik, 2000) are subsequently employed on the new point sets to retrieve partitions.

These methods also vary in the number of eigenvectors they use and the geometrical representation they adopt. Some take K eigenvectors to construct 2^K partitions using recursive bipartitioning (Shi and Malik, 2000) or hypercube partitioning (Hendrickson and Leland, 1995). To get K flat partitions, most prior work uses K eigenvectors (Chan et al., 1994; Shi and Malik, 2000; Weiss, 1999; Ng et al., 2002), others use more eigenvectors than partitions required (Alpert et al., 1995; Malik et al., 2001). Sometimes, the first trivial eigenvector is discarded (Hall, 1970; Shi and Malik, 1997). Some works use eigenvectors literally as point coordinates (Shi and Malik, 2000; Alpert and Kahng, 1995b), while many normalize these points to have unit lengths to construct interpretable affinity matrices (Scott and Longuet-Higgins, 1991; Chan et al., 1994), or with justifications from perturbation theory (Shi and Malik, 1998; Ng et al., 2002).

We will show that such heuristic post-processing can be avoided through a clearer understanding of the relationships between multiple eigenvectors and multiclass partitioning criteria. In fact, we show that such heuristics, which bring in unnecessary assumptions, are not needed. The eigenvectors are more than a geometric embedding. They completely characterize the structure of all optimal solutions we are seeking. Unlike most search-based optimization methods which give only one instance of optima at a time, these eigenvectors provide a compass for us to navigate the whole space of global optima. In the neighbourhoods of these continuous optima, we can discover many near global-optimal discrete solutions that suit our needs.

In this chapter, we will detail how to obtain such a near-global optimum in a principled manner. We first generalize the bipartitioning-based normalized cuts criterion (Shi and Malik, 2000) to multiclass problems. Then we develop its computational solution. Illustrated in Fig 2.4, our method has two stages. (1) We solve a relaxed continuous optimization problem. The set of global optima in a transformed space are generated by a set of eigenvectors Z^*, with arbitrary orthogonal transforms R. Each of them corresponds to an optimal partitioning $\tilde{X}^* R$ in the continuous domain. (2) We obtain a discrete solution X^* closest to the set of continuous optima $\tilde{X}^* R$. This is done iteratively by an alternating optimization procedure: given a discrete solution, we solve for its closest continuous optimum; given a continuous solution, we solve for its closest discrete partitioning solution. After convergence, X^* corresponds to a partitioning that is nearly globally optimal. Finally we illustrate our ideas on the point set data and show our results on real image segmentation.

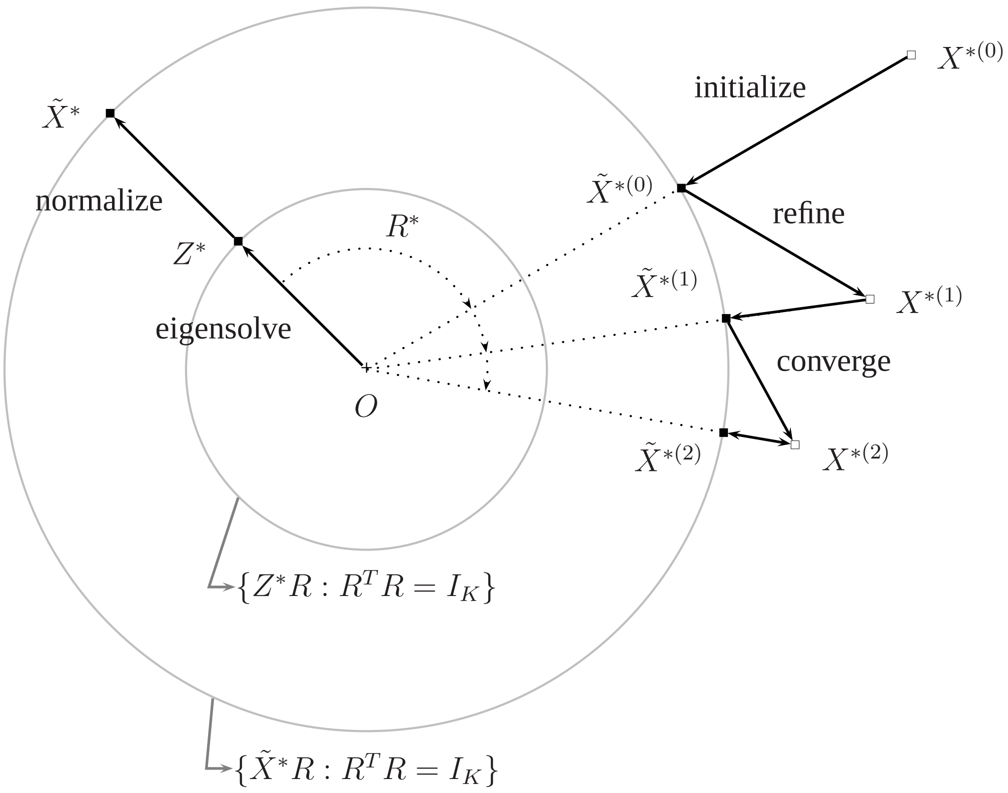


Figure 2.4: Schematic diagram of our algorithm. Here, each point represents a high-dimensional partitioning solution (not to be confused with the 2D phase-plots in Fig 2.6). $(O, +)$: origin of this space. Inner and outer circles: sets of continuous optima for transformed and original representations of partitioning solutions. R: any orthogonal transform. (X^*,□): discrete solutions. ($\tilde{X}^*$,■): continuous solutions. Stage #1: we obtain a global optimum for the transformed representation by eigenvectors Z^*. It sweeps out an orbit of all global optima using orthogonal transform R. They are mapped back to the original representation with their lengths normalized. Stage #2: we obtain a discrete solution closest the orbit of the continuous optima. Given a point A, its closest point on the orbit is the intersection of the line OA and the circle. Therefore, starting from $X^{*(0)}$, we find its closest continuous optimum by computing the best orthogonal transform R^* to bring $\tilde{X}^*$ to $\tilde{X}^{*(0)}$. Among all feasible discrete solutions, $X^{*(1)}$ is the closest to $\tilde{X}^{*(0)}$. Again, we re-compute R^* to bring $\tilde{X}^*$ to $\tilde{X}^{*(1)}$, whose nearest discrete neighbour becomes $X^{*(2)}$. The closest continuous optimum to $X^{*(2)}$ is $\tilde{X}^{*(2)}$, whose nearest discrete neighbour is still $X^{*(2)}$. The algorithm converges. $X^{*(2)}$ is nearly global-optimal.

2.1 Multiclass Normalized Cuts

A weighted graph is specified by $\mathbb{G} = (\mathbb{V}, \mathbb{E}, W)$, where $\mathbb{V}$ is the set of all nodes; $\mathbb{E}$ is the set of edges connecting the nodes; W is an affinity matrix, with weights characterizing the likelihood that two nodes belong to the same group. W is assumed nonnegative and symmetric.

Let $[n]$ denote the set of integers between 1 and n: $[n] = \{1, 2, \ldots, n\}$. Let $\mathbb{V} = [N]$ denote the set of all elements (data points or pixels) to be grouped. To cluster N points into K groups is to decompose $\mathbb{V}$ into K disjoint sets, i.e., $\mathbb{V} = \cup_{l=1}^{K} \mathbb{V}_l$ and $\mathbb{V}_k \cap \mathbb{V}_l = \varnothing$, $k \neq l$. We denote this K-way partitioning by $\Gamma_{\mathbb{V}}^{K} = \{\mathbb{V}_1, \ldots, \mathbb{V}_K\}$.

2.1.1 Multiclass Partitioning Criteria

Let node sets $\mathbb{P}, \mathbb{Q} \subset \mathbb{V}$. We define $\mathrm{links}(\mathbb{P}, \mathbb{Q})$ to be the total weighted connections from $\mathbb{P}$ to $\mathbb{Q}$:

$$\mathrm{links}(\mathbb{P}, \mathbb{Q}) = \sum_{p \in \mathbb{P},\, q \in \mathbb{Q}} W(p, q). \tag{2.2}$$

The degree of a set is its total connections to all the nodes:

$$\mathrm{degree}(\mathbb{P}) = \mathrm{links}(\mathbb{P}, \mathbb{V}). \tag{2.3}$$

Using the degree as a normalization term, we define $\mathrm{linkratio}(\mathbb{P}, \mathbb{Q})$ as the *proportion* of the connections from $\mathbb{P}$ to $\mathbb{Q}$ among all those $\mathbb{P}$ has:

$$\mathrm{linkratio}(\mathbb{P}, \mathbb{Q}) = \frac{\mathrm{links}(\mathbb{P}, \mathbb{Q})}{\mathrm{degree}(\mathbb{P})}. \tag{2.4}$$

Two special $\mathrm{linkratio}$'s are of particular interest:

$\mathrm{linkratio}(\mathbb{P}, \mathbb{P})$:	measures how many links *stay* within $\mathbb{P}$ itself;
$\mathrm{linkratio}(\mathbb{P}, \mathbb{V} \setminus \mathbb{P})$:	measures how many links *escape* from $\mathbb{P}$.

A good clustering desires both tight connections within partitions and loose connections between partitions (Fig 2.5). This intuition is quantitatively captured by

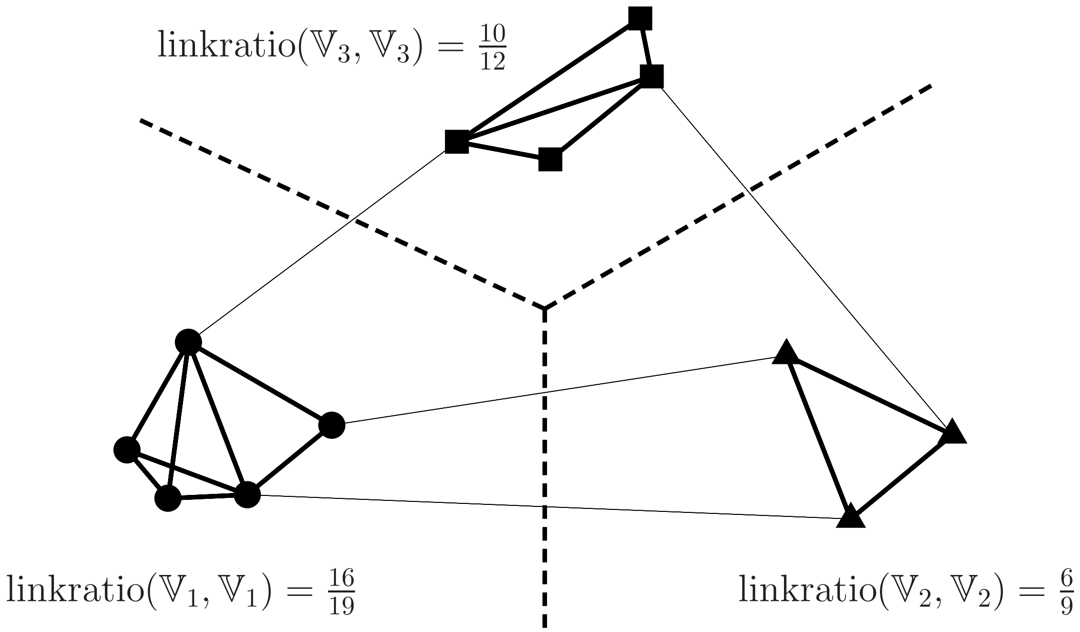

$\mathrm{linkratio}(\mathbb{V}_3, \mathbb{V}_3) = \frac{10}{12}$
$\mathrm{linkratio}(\mathbb{V}_1, \mathbb{V}_1) = \frac{16}{19}$
$\mathrm{linkratio}(\mathbb{V}_2, \mathbb{V}_2) = \frac{6}{9}$

Figure 2.5: K-way normalized cuts criterion. Consider the above 3-way partitioning of nodes into ●, ▲, ■ groups. Edges intersecting the dashed lines are those being cut by the partitioning. The three $\mathrm{linkratio}$'s are given in the figure, assuming that each edge has a unit weight. Note that an edge is counted twice for within-group connections. A good partitioning has most (thick) edges contained in each group, allowing as few (thin) edges escaping as possible.

large $\mathrm{linkratio}(\mathbb{P}, \mathbb{P})$ and small $\mathrm{linkratio}(\mathbb{P}, \mathbb{V} \setminus \mathbb{P})$. Thanks to the normalization, these two goals are in no conflict, as:

$$\mathrm{linkratio}(\mathbb{P}, \mathbb{P}) + \mathrm{linkratio}(\mathbb{P}, \mathbb{V} \setminus \mathbb{P}) = 1. \tag{2.5}$$

Generalizing this idea on a single node set $\mathbb{P}$ to K node sets that make up $\mathbb{V}$, we are able to formulate a goodness measure for a K-way partitioning. Formally, we define K-way *normalized associations* and *normalized cuts* criteria:

$$\mathrm{knassoc}(\Gamma_{\mathbb{V}}^{K}) = \frac{1}{K} \sum_{l=1}^{K} \mathrm{linkratio}(\mathbb{V}_l, \mathbb{V}_l) \tag{2.6}$$

$$\mathrm{kncuts}(\Gamma_{\mathbb{V}}^{K}) = \frac{1}{K} \sum_{l=1}^{K} \mathrm{linkratio}(\mathbb{V}_l, \mathbb{V} \setminus \mathbb{V}_l). \tag{2.7}$$

Clearly $\mathrm{knassoc}(\Gamma_{\mathbb{V}}^{K}) + \mathrm{kncuts}(\Gamma_{\mathbb{V}}^{K}) = 1$, i.e. maximizing the associations and minimizing the cuts are achieved simultaneously.

Among numerous partitioning criteria such as minimum cuts and various definitions of average cuts, only minimum cuts and normalized cuts have this duality property. However, minimum cuts are noise-sensitive, i.e., a few isolated nodes could easily draw the cuts away from a global partitioning, whereas normalized cuts are robust to weight noise (Shi and Malik, 2000). Since $\mathrm{knassoc}$ and kncuts are equivalent, we make no distinction further and denote our objective as:

$$\varepsilon(\Gamma_{\mathbb{V}}^{K}) = \mathrm{knassoc}(\Gamma_{\mathbb{V}}^{K}). \tag{2.8}$$

ε is a unit-less value between 0 and 1 regardless of K.

For any K-way partitioning criterion, we need to examine its performance over K's: how does it change with K? Can it produce a refinement of partitioning when K increases? The definitions in Eqn (2.6) and (2.7) do not lend an obvious answer to these questions. However, we will show that an upperbound of ε decreases monotonically with increasing K. Although ε itself does not entail the requirement of hierarchical refinement over the number of classes, a consistent optimal partitioning can often be obtained with little extra cost.

2.1.2 Representation

We define the *degree matrix* for the symmetric weight matrix W to be:

$$D = \mathrm{Diag}(W 1_N), \tag{2.9}$$

where Diag forms a diagonal matrix from its vector argument and 1_d denotes the $d \times 1$ vector of all 1's. We use $N \times K$ *partition matrix* X to represent $\Gamma_{\mathbb{V}}^{K}$. Let $X = [X_1, \ldots, X_K]$, where X_l is a binary indicator for $\mathbb{V}_l$:

$$X(i, l) = \mathrm{istrue}(i \in \mathbb{V}_l), \quad i \in \mathbb{V},\ l \in [K], \tag{2.10}$$

where $\mathrm{istrue}(\cdot)$ is 1 if the argument is true and 0 otherwise. Since a node is assigned to one and only one partition, there is an exclusion constraint between columns of X: $X\,1_K = 1_N$.

With these symbols, we rewrite links and degree as functions of X:

$$\begin{aligned} \mathrm{links}(\mathbb{V}_l, \mathbb{V}_l) &= X_l^T W X_l & (2.11) \\ \mathrm{degree}(\mathbb{V}_l) &= X_l^T D X_l. & (2.12) \end{aligned}$$

The K-way normalized cuts criterion is expressed in an optimization program of variable X, called program *PNCX*:

$$\text{maximize} \quad \varepsilon(X) = \frac{1}{K} \sum_{l=1}^{K} \frac{X_l^T W X_l}{X_l^T D X_l} \qquad (2.13)$$

$$\text{subject to} \quad X \in \{0, 1\}^{N \times K}, \quad X\, 1_K = 1_N. \qquad (2.14)$$

This problem is NP-complete even for a planar graph at $K = 2$ (Shi and Malik, 2000). We will develop a fast algorithm to find its near-global optima.

2.2 Solving K-way Normalized Cuts

We solve program *PNCX* in two steps. We first relax a transformed formulation into an eigenvalue problem. We show that its global optimum is not unique, and a special solution is eigenvectors of (W, D). Transforming the eigenvectors to the space of partition matrices, we get a set of continuous global optima. We then solve a discretization problem, where the discrete partition matrix closest to the continuous optima is sought. Such a discrete solution is thus near global-optimal.

2.2.1 Finding Optimal Relaxed Solutions

We introduce *scaled partition matrix* Z to make Eqn (2.13) more manageable. Let

$$Z = X(X^T D X)^{-\frac{1}{2}}. \qquad (2.15)$$

Since $X^T D X$ is diagonal, the columns of Z are simply those of X scaled by the inverse square root of the degrees of partitions. We then have $\varepsilon(X) = \frac{1}{K}\,\mathrm{tr}(Z^T W Z)$, where tr denotes the trace of a matrix. A natural constraint on Z is:

$$Z^T D Z = (X^T D X)^{-\frac{1}{2}} X^T D X (X^T D X)^{-\frac{1}{2}} = I_K,$$

where I_K denotes the $K \times K$ identity matrix. Ignoring the constraints in *PNCX*, we derive a new program of variable Z and call it *PNCZ*:

$$\begin{aligned} \text{maximize} \quad & \varepsilon(Z) = \frac{1}{K} \operatorname{tr}(Z^T W Z) && (2.16) \\ \text{subject to} \quad & Z^T D Z = I_K. && (2.17) \end{aligned}$$

Relaxing Z into the continuous domain turns the discrete problem into a tractable continuous optimization problem. The special structure of this program is revealed in Proposition 1, which can be proved trivially using $\operatorname{tr}(AB) = \operatorname{tr}(BA)$.

Proposition 1 (Orthogonal Invariance). *Let R be a $K \times K$ matrix. If Z is a feasible solution to* PNCZ*, so is $\{ZR : R^T R = I_K\}$. Furthermore, they have the same objective value: $\varepsilon(ZR) = \varepsilon(Z)$.*

Therefore, a feasible solution remains equally good with arbitrary rotation and reflection. Program *PNCZ* is a Rayleigh quotient optimization problem that has been addressed in Rayleigh-Ritz theorem and its extensions. Proposition 2 rephrases the theorem in our problem setting. It can also be proven directly using Lagrangian relaxation. The proposition shows that among all the optima are the eigenvectors of (W, D), or equivalently those of *normalized weight matrix* P:

$$P = D^{-1} W. \tag{2.18}$$

Since P is a stochastic matrix (Meila and Shi, 2001), it is easy to verify that 1_N is a trivial eigenvector of P and it corresponds to the largest eigenvalue of 1.

Proposition 2 (Optimal Eigensolution). *Let (V, S) be the eigendecomposition of P: $PV = VS$, where $V = [V_1, \ldots, V_N]$ and $S = \operatorname{Diag}(s)$ with eigenvalues ordered nonincreasingly: $s_1 \geq \ldots \geq s_N$. (V, S) is obtained from the orthonormal eigensolution $(\bar{V}, S)$ of the symmetric matrix $D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$, where*

$$\begin{aligned} V &= D^{-\frac{1}{2}} \bar{V}, && (2.19) \\ D^{-\frac{1}{2}} W D^{-\frac{1}{2}} \bar{V} &= \bar{V} S, \quad \bar{V}^T \bar{V} = I_N. && (2.20) \end{aligned}$$

Therefore, V and S are all real and any K distinct eigenvectors form a local optimum candidate to PNCZ, *with*

$$\varepsilon([V_{\pi_1}, \dots, V_{\pi_K}]) = \frac{1}{K} \sum_{l=1}^{K} s_{\pi_l}, \tag{2.21}$$

where π is an index vector of K distinct integers from $[N]$. The global optimum of PNCZ *is thus achieved when $\pi = [1, \dots, K]$:*

$$\begin{aligned} Z^* &= [V_1, \dots, V_K], & (2.22) \\ \Lambda^* &= \mathrm{Diag}([s_1, \dots, s_K]), & (2.23) \\ \varepsilon(Z^*) &= \frac{1}{K} \mathrm{tr}(\Lambda^*) = \max_{Z^T D Z = I_K} \varepsilon(Z). & (2.24) \end{aligned}$$

Strictly speaking, any distinct K columns of V are locally optimal only if the weight matrix is semi-positive definite. Though a pairwise weight matrix evaluated from a Gaussian function is guaranteed to be positive definite (Berg et al., 1984), which is what we will use in our experiments, in general it may not be semi-positive definite. In other words, the objective function may not be convex.

However, the global-optimality holds even when it is non-convex. The Rayleigh quotient is known to play an important role in the optimization of a non-convex function on a possibly nonconvex set. Our relaxation technique belongs to the family of trust region subproblems, where strong duality (zero duality gap) holds for a larger class of seemingly non-convex problems. A rigorous exposition can be found in (Anstreicher and Wolkowicz, 2000) and references therein.

To summarize, the global optimum of *PNCZ* is not unique. It is a subspace spanned by the first K largest eigenvectors of P through orthogonal matrices:

$$\{Z^* R : \ R^T R = I_K, \ P Z^* = Z^* \Lambda^*\}. \tag{2.25}$$

Unless the eigenvalues are all the same, $Z^* R$ are no longer the eigenvectors of P. All these solutions have the optimal objective value, which provides a nonincreasing upperbound to *PNCX*.

Corollary 1 (Upperbound Monotonicity). *For any K,*

$$\max \varepsilon(\Gamma_{\mathbb{V}}^{K}) \leq \max_{Z^T D Z = I_K} \varepsilon(Z) = \frac{1}{K}\sum_{l=1}^{K} s_l \tag{2.26}$$

$$\max_{Z^T D Z = I_{K+1}} \varepsilon(Z) \leq \max_{Z^T D Z = I_K} \varepsilon(Z). \tag{2.27}$$

Next we transform Z back to the space of partition matrices. If f is the mapping that scales X to Z, then f^{-1} is the normalization that brings Z back to X:

$$Z = f(X) = X(X^T D X)^{-\frac{1}{2}} \tag{2.28}$$

$$X = f^{-1}(Z) = \mathrm{Diag}(\mathrm{diag}^{-\frac{1}{2}}(Z Z^T))\, Z, \tag{2.29}$$

where diag returns the diagonal of its matrix argument in a column vector. If we take the rows of Z as coordinates of K-dimensional points, what f^{-1} does is to *normalize* their lengths so that they lie on the unit hypersphere centered at the origin. With f^{-1}, we transform the continuous optimum $Z^* R$ in the Z-space to the X-space: since $R^T R = I_K$,

$$f^{-1}(Z^* R) = f^{-1}(Z^*) R. \tag{2.30}$$

This simplification is important because now the continuous optima are directly characterized by $f^{-1}(Z^*)$ in the X-space:

$$\{\tilde{X}^* R :\ \tilde{X}^* = f^{-1}(Z^*),\ R^T R = I_K\}. \tag{2.31}$$

2.2.2 Special Case: Bipartitioning

Before we proceed to discretize the continuous optima, we examine 2-class normalized cuts in our multiclass framework. Instead of transforming the eigenvectors to approach an underlying discrete solution, we transform the underlying solution to approach the eigenvectors. By doing so, we reach a direct interpretation of eigenvectors.

Proposition 3. *Let $d = 1^T D 1$ denote the total degree of all the nodes in the graph. Let $\alpha = X_1^T D X_1 / d$ denote the degree ratio of the first group $\mathbb{V}_1$. For $K = 2$, there exists an orthogonal matrix R such that the first column of ZR = $X(X^T DX)^{-\frac{1}{2}} R$ is a multiple of 1_N:*

$$R = \begin{bmatrix} \sqrt{\alpha} & -\sqrt{1-\alpha} \\ \sqrt{1-\alpha} & \sqrt{\alpha} \end{bmatrix}, \tag{2.32}$$

$$ZR = \frac{1}{\sqrt{d}} \cdot \left[1_N \quad \sqrt{\tfrac{\alpha}{1-\alpha}} - \sqrt{\tfrac{1}{\alpha(1-\alpha)}} \cdot X_1 \right] \tag{2.33}$$

Proof. Since $X_1 + X_2 = 1_N$ and $X^T DX = d \cdot \mathrm{Diag}([\alpha, 1-\alpha])$, we have:

$$\begin{aligned} ZR &= X(X^T DX)^{-\frac{1}{2}} R \\ &= \frac{1}{\sqrt{d}} \cdot \left[\tfrac{X_1}{\sqrt{\alpha}} \quad \tfrac{1-X_1}{1-\alpha} \right] \cdot \begin{bmatrix} \sqrt{\alpha} & -\sqrt{1-\alpha} \\ \sqrt{1-\alpha} & \sqrt{\alpha} \end{bmatrix} \\ &= \frac{1}{\sqrt{d}} \cdot \left[1_N \quad \sqrt{\tfrac{\alpha}{1-\alpha}} - \left(\sqrt{\tfrac{\alpha}{1-\alpha}} + \sqrt{\tfrac{1-\alpha}{\alpha}} \right) \cdot X_1 \right]. \end{aligned}$$

Reduction of the last equation gives Eqn (2.33). □

Since 1_N is an eigenvector of P, we immediately see that the second eigenvector of P is an approximation to the linear function of X_1 given in the second column of Eqn (2.33). It is a scaled version of the variable used in (Shi and Malik, 2000) for determining the solution to 2-class normalized cuts. To make it more explicit, given R in Eqn (2.32), let $Y = ZR = [Y_1, Y_2]$. The reader can verify that $Y^T DY = I_2$ is automatically satisfied. Let $y = X_1 - \alpha \cdot 1_N$, i.e. $Y_2 = \beta \cdot y$, where $\beta = \sqrt{d\alpha(1-\alpha)}$. Then we have:

$$\varepsilon(Y) = \frac{1}{2} \operatorname{tr}(Y^T WY) = \frac{1}{2} \left(\frac{1}{d} 1_N^T W 1_N + Y_2^T W Y_2 \right) = \frac{1}{2} \left(1 + \beta^2 y^T W y \right),$$

$$y^T D y = \frac{1}{\beta^2} Y_2^T D Y_2 = \frac{1}{\beta^2}.$$

Based on the definition on α, there is a natural constraint on y: $y^T D 1_N = 0$. For 2-class problems, we reduce *PNCZ* to a Rayleigh quotient in the single scaled indicator y:

$$\text{maximize} \quad \varepsilon(y;W) = \frac{1}{2}\left(\frac{y^T W y}{y^T D y} + 1\right), \tag{2.34}$$

$$\text{subject to} \quad y^T D 1_N = 0. \tag{2.35}$$

The question that follows is this: if the second eigenvector is a relaxed version of $X_1 - \alpha \cdot 1_N$ or equivalently $(1-\alpha) \cdot X_1 - \alpha X_2$, what about the third and fourth eigenvector? Even if we have these eigenvectors, how do we recover X from these transformations?

These are the difficulties involved when generalizing such a reduction procedure to $K > 2$. It is a major reason why multiclass normalized cuts have not been studied formally. Extrapolating recklessly to multiclass problems could be misleading: a hierarchical segmentation was suggested (Shi and Malik, 2000), i.e., 2^K partitions only need K eigenvectors, with one subsequent eigenvector for a successive bipartitioning. It is now clear that we need K and only K eigenvectors to yield K (not 2^K) partitions. The reason is that group indicators are constrained to be orthogonal. They cannot be chosen freely, as required for hierarchical cuts.

We also gain more perspective on the first eigenvector. Though $Z_1^* = d^{-\frac{1}{2}} \cdot 1_N$ is a trivial multiple of 1_N, $\tilde{X}_1^*$ is not for $K > 1$. The seemingly trivial first eigenvector is as important as any others, since they collectively provide a basis for generating the whole set of continuous solutions that optimize the objective.

2.2.3 Finding Optimal Discrete Solutions

The optimal solutions to *PNCZ* are in general not feasible to the original program *PNCX*. However, we can use them to find a nearby discrete solution. This discrete solution may not be the absolute maximizer of *PNCX*, but it is nearly global-optimal due to the continuity of the objective function. Therefore, our goal here is to find a discrete solution that satisfies the binary constraints of the original program *PNCX*, yet is closest to the continuous optima given in Eqn (2.31).

Theorem 1 (Optimal Discretization). *Let $\tilde{X}^* = f^{-1}(Z^*)$. An optimal discrete* *partition X^* is considered the one satisfying the following program called* POD*:*

$$\begin{aligned} \textit{minimize} \quad & \phi(X,R) = \|X - \tilde{X}^* R\|^2 && (2.36)\\ \textit{subject to} \quad & X \in \{0,1\}^{N\times K}, \quad X\,1_K = 1_N && (2.37)\\ & R^T R = I_K, && (2.38)\end{aligned}$$

where $\|M\|$ denotes the Frobenius norm of matrix M: $\|M\| = \sqrt{\mathrm{tr}(MM^T)}$. A local optimum (X^*, R^*) of this bilinear program can be solved iteratively.

Given R^*, POD *is reduced to program* PODX *in X:*

$$\begin{aligned} \textit{minimize} \quad & \phi(X) = \|X - \tilde{X}^* R^*\|^2 && (2.39)\\ \textit{subject to} \quad & X \in \{0,1\}^{N\times K}, \quad X\,1_K = 1_N. && (2.40)\end{aligned}$$

Let $\tilde{X} = \tilde{X}^* R^*$. The optimal solution is given by non-maximum suppression (if there are multiple maxima, only one of them, but any one of them, can be chosen so as to honor the exclusion constraint on a partition matrix):

$$X^*(i,l) = \mathrm{istrue}(l = \arg\max_{k\in[K]} \tilde{X}(i,k)), \quad i \in \mathbb{V}. \qquad (2.41)$$

Given X^*, POD *is reduced to program* PODR *in R:*

$$\begin{aligned} \textit{minimize} \quad & \phi(R) = \|X^* - \tilde{X}^* R\|^2 && (2.42)\\ \textit{subject to} \quad & R^T R = I_K, && (2.43)\end{aligned}$$

and the solution is given through some singular vectors:

$$\begin{aligned} R^* &= \tilde{U} U^T, && (2.44)\\ X^{*T} \tilde{X}^* &= U \Omega \tilde{U}^T, \quad \Omega = \mathrm{Diag}(\omega), && (2.45)\end{aligned}$$

where $(U, \Omega, \tilde{U})$ is a singular value decomposition (SVD) of $X^{*T}\tilde{X}^*$, with $U^T U = I_K$, $\tilde{U}^T \tilde{U} = I_K$ and $\omega_1 \geq \ldots \geq \omega_K$.

Proof. First we note that: $\phi(X,R) = \|X\|^2 + \|\tilde{X}^*\|^2 - \mathrm{tr}(XR^T\tilde{X}^{*T} + X^T\tilde{X}^*R) = 2N - 2\,\mathrm{tr}(XR^T\tilde{X}^{*T})$. Thus minimizing $\phi(X,R)$ is equivalent to maximizing $\mathrm{tr}(XR^T\tilde{X}^{*T})$. For *PODX*, given $R = R^*$, as each entry of $\mathrm{diag}(XR^{*T}\tilde{X}^{*T})$

can be optimized independently, Eqn (2.41) results. For *PODR*, given $X = X^*$, we construct a *Lagrangian* using a symmetric matrix multiplier Λ:

$$L(R, \Lambda) = \operatorname{tr}(X^* R^T \tilde{X}^{*T}) - \frac{1}{2} \operatorname{tr}(\Lambda^T (R^T R - I_K)).$$

The optimum (R^*, Λ^*) must satisfy

$$L_R = \tilde{X}^{*T} X^* - R\Lambda = 0, \quad \text{i.e.} \quad \Lambda^* = R^{*T} \tilde{X}^{*T} X^*. \tag{2.46}$$

Thus $\Lambda^{*T}\Lambda^* = U\Omega^2 U^T$. Since $\Lambda = \Lambda^T$, $\Lambda^* = U\Omega U^T$. From Eqn (2.46), we then have: $R^* = \tilde{U} U^T$ and $\phi(R^*) = 2N - 2\operatorname{tr}(\Omega)$. The larger $\operatorname{tr}(\Omega)$ is, the closer X^* is to $\tilde{X}^* R^*$. □

Due to the orthogonal invariance of the continuous optima, our method is robust to arbitrary initialization, from either X or R. A good initialization can nevertheless speed up convergence. We find the heuristic mentioned in (Ng et al., 2002) is good and fast. It is simply K-means clustering with K nearly orthogonal data points as centers. Computationally, it is equivalent to initialize R^* by choosing K rows of $\tilde{X}^*$ that are as orthogonal to each other as possible. To derive X^* by Eqn (2.41) on this non-orthogonal R^* is exactly K-means clustering with the unit-length centers.

Given X^*, we solve *PODR* to find a continuous optimum $\tilde{X}^* R^*$ closest to it. For this continuous optimum, we then solve *PODX* to find its closest discrete solution. Each step reduces the same objective ϕ through coordinate descent. We can only guarantee such iterations end in a local optimum, which may vary with the initial estimation. However, since $\tilde{X}^* R^*$ are all globally optimal regardless of R^*, whichever $\tilde{X}^* R^*$ the program *POD* converges to, its proximal discrete solution X^* will not be too much off from the optimality.

The loss of optimality in discretization might result in $\varepsilon(X^*) < \frac{1}{K}\sum_{l=2}^{K+1} s_l$, which is the next optimal value in the continuous domain and is achieved by $Z = [V_2, \ldots, V_{K+1}]$. A legitimate concern is whether a nearby discrete solution X of this suboptimal Z would suffer less loss in discretization and become optimal: $\varepsilon(X) > \varepsilon(X^*)$. We don't have any theoretical analysis regarding this issue, but such reversal of optimality is rare if ever in our experiments.

2.2.4 Algorithm

Given weight matrix W and desired number of classes K:

1. Compute the degree matrix $D = \mathrm{Diag}(W 1_N)$.
2. Find the optimal eigensolution Z^* by:

   $$\begin{aligned} &D^{-\frac{1}{2}} W D^{-\frac{1}{2}}\, \bar{V}_{[K]} = \bar{V}_{[K]}\, \mathrm{Diag}(s_{[K]}), \quad \bar{V}_{[K]}^T \bar{V}_{[K]} = I_K \\ &Z^* = D^{-\frac{1}{2}} \bar{V}_{[K]}. \end{aligned}$$

3. Normalize Z^* by: $\tilde{X}^* = \mathrm{Diag}(\mathrm{diag}^{-\frac{1}{2}}(Z^* Z^{*T})) Z^*$.
4. Initialize X^* by computing R^* as:

   $$\begin{aligned} &R_1^* = [\tilde{X}^*(i,1), \ldots \tilde{X}^*(i,K)]^T, \text{ random } i \in [N] \\ &c = 0_{N \times 1} \\ &\text{For } k = 2, \ldots, K, \text{ do:} \\ &\qquad c = c + \mathrm{abs}(\tilde{X}^* R_{k-1}^*) \\ &\qquad R_k^* = [\tilde{X}^*(i,1), \ldots \tilde{X}^*(i,K)]^T, \; i = \arg\min c \end{aligned}$$

5. Initialize convergence monitoring parameter $\bar{\phi}^* = 0$.
6. Find the optimal discrete solution X^* by:

   $$\begin{aligned} &\tilde{X} = \tilde{X}^* R^* \\ &X^*(i,l) = \mathrm{istrue}(l = \arg\max_{k \in [K]} \tilde{X}(i,k)), \quad i \in \mathbb{V},\; l \in [K]. \end{aligned}$$

7. Find the optimal orthogonal matrix R^* by:

   $$\begin{aligned} &X^{*T} \tilde{X}^* = U \Omega \tilde{U}^T, \quad \Omega = \mathrm{Diag}(\omega) \\ &\bar{\phi} = \mathrm{tr}(\Omega) \\ &\text{If } |\bar{\phi} - \bar{\phi}^*| < \text{machine precision, then stop and output } X^* \\ &\bar{\phi}^* = \bar{\phi} \\ &R^* = \tilde{U} U^T \end{aligned}$$

8. Go to Step 6.

In Step 2, we use $\bar{V}_{[K]}$ as a shorthand for $[\bar{V}_1, \ldots, \bar{V}_K]$, and likewise for $\bar{S}_{[K]}$. In Step 4, $B = \mathrm{abs}(A)$ denotes the absolute values of the elements of A. In Step 3, since $\tilde{X}^* = \mathrm{Diag}(\mathrm{diag}^{-\frac{1}{2}}(Z^* Z^{*T}))Z^*$ scales the lengths of each row to 1, we can skip scaling $\bar{V}$ in order to get V, i.e. $Z^* = [\bar{V}_1, \ldots, \bar{V}_K]$ leads to the same $\tilde{X}^*$.

Step 2 solves the first K leading eigenvectors of an $N \times N$ usually sparse matrix. It is nevertheless the most time consuming, with a time complexity of $O(N^{\frac{3}{2}}K)$ using a Lancoz eigensolver in our image segmentation experiments. See an analysis in (Shi, 1998). Step 4 has $NK(K-1)$ multiplications in choosing K centers. Step 6 involves NK^2 multiplications to compute $\tilde{X}^* R^*$. Step 7 involves an SVD of a $K \times K$ matrix and K^3 multiplications for computing R^*. Since X^* is binary, $X^{*T}\tilde{X}^*$ can be done efficiently with all additions. Taken together, the time complexity of the algorithm is $O(N^{\frac{3}{2}}K + NK^2)$.

2.3 Experiments

For the point set used in the introduction, we have shown the four eigenvectors of P in Fig 2.3. These eigenvectors provide the basis for up to 4-class partitioning. Using this point set data, we illustrate the flow of our algorithm in Fig 2.6. An example using suboptimal continuous solutions is given in Fig 2.7. The results for $K > 2$ are given in Fig 2.8. There is loss of optimality in discrete partitions, but they are nearly global-optimal.

Images are first convolved with oriented filter pairs to extract the magnitude of edge responses OE (Malik et al., 2001). Pixel affinity W is inversely correlated with the maximum magnitude of edges crossing the line connecting two pixels. $W(i,j)$ is low if i, j are on the two sides of a strong edge (Fig 2.9):

$$W(i,j) = \exp\left(-\frac{\max_{t \in (0,1)} OE^2(\underline{i} + t \cdot \underline{j})}{2\sigma_e^2 \cdot \max_k OE^2(\underline{k})}\right), \tag{2.47}$$

where $\underline{i}$ denotes the location of pixel i. This measure is meaningful only for nearby pixels. We hence set $W(i,j) = 0$ beyond a city-block distance r_W. We fix $\sigma_e = 0.01$ and $r_W = 8$ for all images.

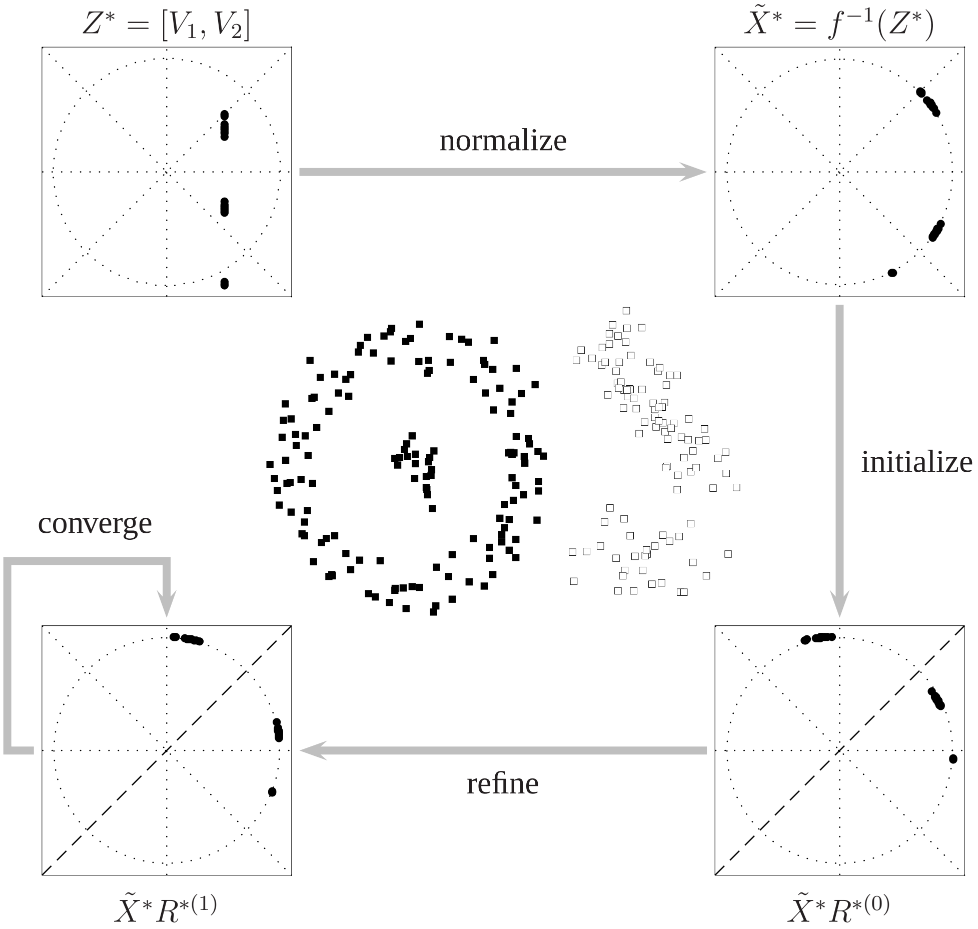


Figure 2.6: Progression of our algorithm. Each plot shows an $N \times 2$ matrix, with each row taken as (x, y) coordinates of a point in the plane. Though there are $N = 245$ points in total, many of them are mapped to the same planar point, resulting in three visible clusters. #1 Normalize: starting with the eigenvectors Z^*, we first map it back to the X-space by normalizing their lengths so that all of them lie on the unit circle. #2 Initialize: two points with almost orthogonal phases are selected to form $R^{*(0)}$. $\tilde{X}^* R^{*(0)}$ is the projection of all the points to the two chosen directions. An initial clustering $X^{*(0)}$ is obtained by non-maximum suppression: points are divided according to the dashed line $x = y$: points below the line assigned to $(1, 0)$ hence $\mathbb{V}_1$, those above the line assigned to $(0, 1)$ hence $\mathbb{V}_2$. #3 Refine: we find the closest continuous optimal to $X^{*(0)}$ by adjusting the rotation matrix $R^{*(1)}$. Non-maximum suppression produces its closest discrete solution $X^{*(1)}$, which is exactly the same as $X^{*(0)}$. The algorithm converges and stops. The final clustering is shown in the center, with $\varepsilon(X^*) = 0.9997 < \varepsilon(\tilde{X}^*) = 0.9998$.

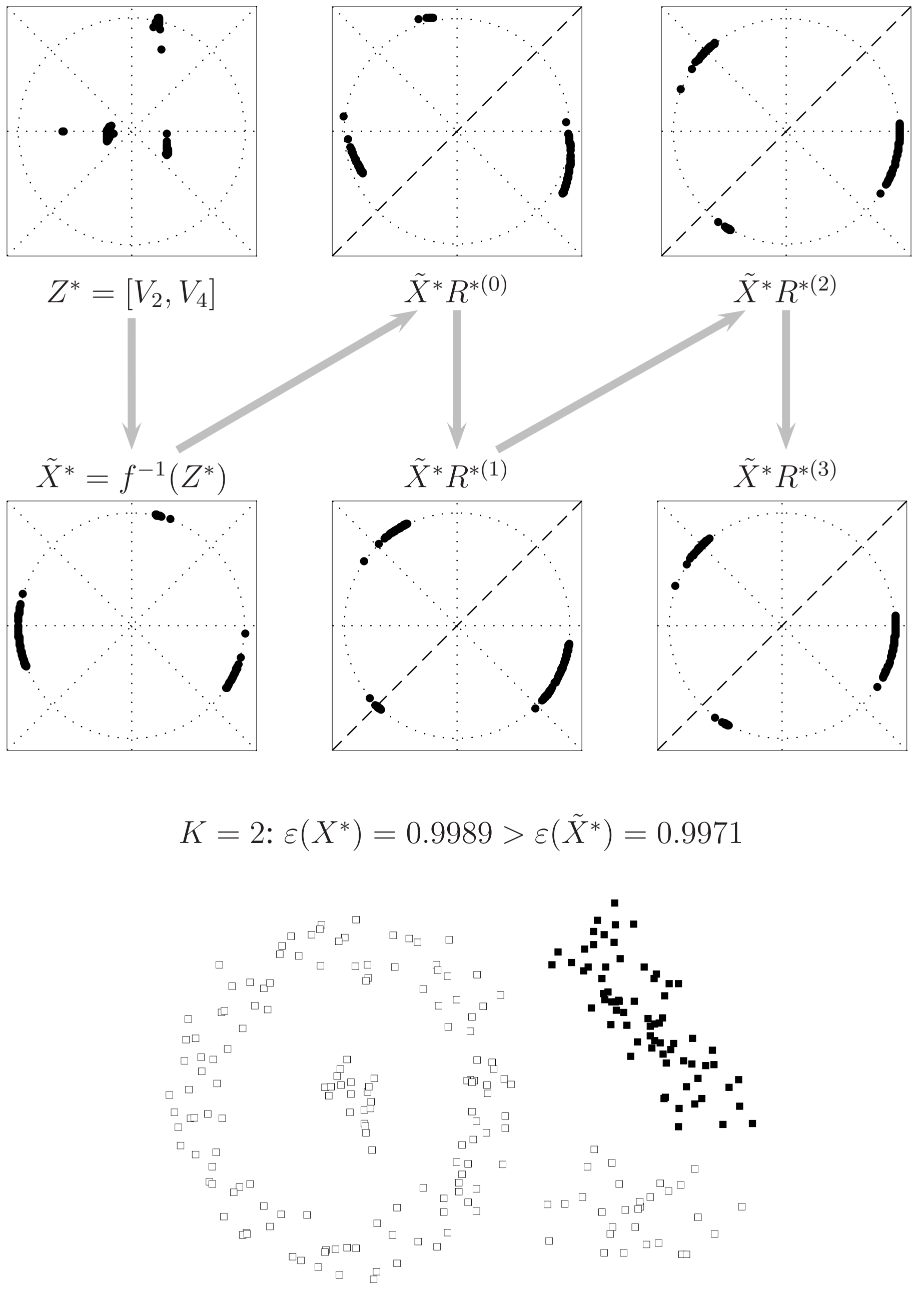
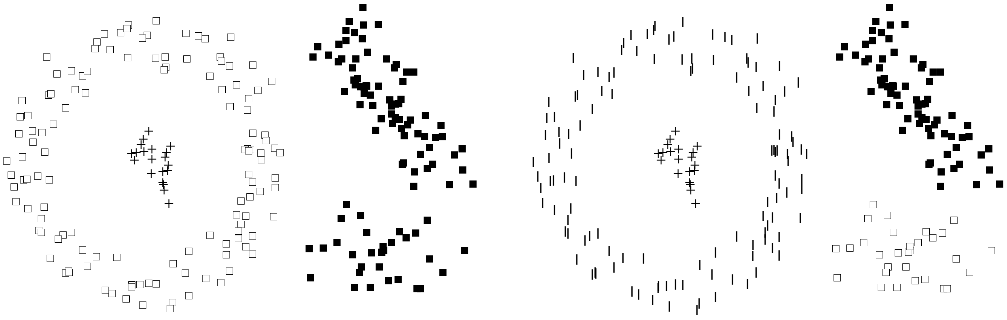


Figure 2.7: Clustering from suboptimal continuous solutions. Same convention as Fig 2.6. After initialization, we refine the clustering by iteratively applying an optimal orthogonal transformation R^*, in this case equivalent to a rotation and a reflection in the plane. After 3 iterations, the algorithm converges to a local optimum where the pair $(X^*, \tilde{X}^* R^*)$ no longer changes. Notice here the discrete solution has a better objective value than its continuous counterpart.



$K = 3$: $(0.9995, 0.9995)$ $K = 4$: $(0.9978, 0.9983)$

Figure 2.8: Clustering from the first 3, 4 eigenvectors. The numbers are $(\varepsilon(X^*), \varepsilon(\tilde{X}^*))$.

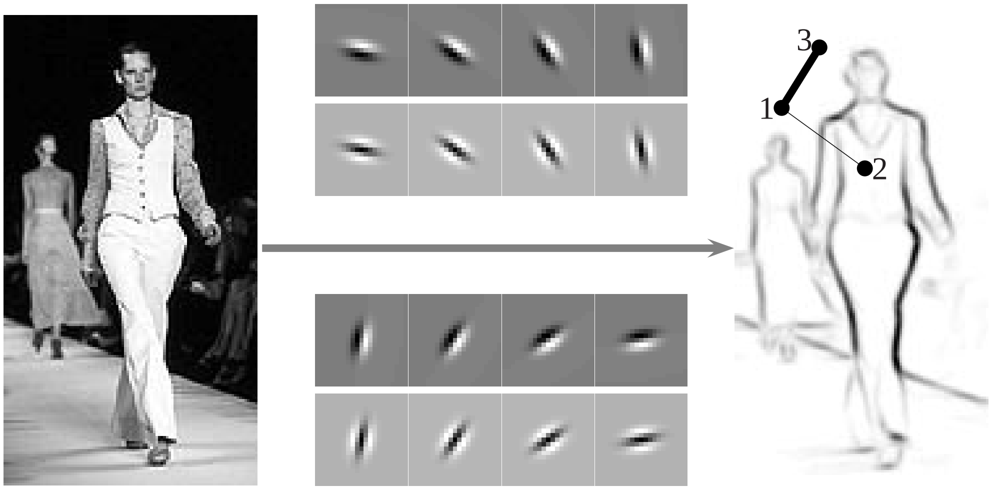


image oriented filter pairs edge magnitudes

Figure 2.9: Pixel affinity matrix W is computed based on intensity edge magnitudes. For example, $W(1,2) \approx 0$ while $W(1,3) \approx 1$.

Real images present a richer structure than the artificial point set data. Fig 2.10 shows the first 20 leading eigenvectors for an image. These eigenvectors can be interpreted as vibration modes in a mass-spring system, with eigenvalues indicating vibration periods. Shown in Fig 2.11, through orthogonal transforms, the coarse-to-fine structure of $\tilde{X}^*$ is folded flat in $\tilde{X}^* R^*$. Nevertheless, refining partitions with increasing K can be achieved through a sequential initialization: we use X^*

of $\Gamma_{\mathbb{V}}^{K}$ as a starting segmentation for $\Gamma_{\mathbb{V}}^{K+1}$, with its largest region broken into 2 groups. This produces a pseudo-hierarchical segmentation in Fig 2.10: when K increases, regions tend to be successively divided (e.g. $K7$, $K8$), yet the enclosing boundaries are subject to fine adjustment (e.g. $K5$, $K6$). Knowing the structure of all continuous optima allows us to pick out a set of good discrete solutions with special specifications.

Our solution to the original normalized cuts formulation *PNCX* is done in two steps: first find continuous optima and then find a closest discrete solution. Although each step is optimal, the output at the end need not be optimal in general. To evaluate the optimality of our solutions as well as to understand the effects of initialization in our discretization step, we collect all possible optimal discrete solutions from our discretization procedure. This is done by initializing the discretization procedure with pixels sampled at a dense grid shown in Fig 2.12.

What is remarkable is that for most K's, i.e. $K = 1$, 2, 3, 6, 7, 8, 9, 12, 13, 17, 18, 19, 20 among $K \leq 20$, all initializations converge to one single optimum (Fig 2.13). For $K = 2$, we show the final continuous solution in Fig 2.12. It is most evident in the phaseplot that there is no other choice for discretization. For other K's, there could be multiple solutions that attract different initializations (Fig 2.14). Despite many discrete solutions in the vicinity of a continuous optimum, only a few of them are *closest* to the *set* of continuous optima. What distinguishes them from other discrete solutions, for example, the intermediate solutions during iterations (Fig 2.11), is that they all have relatively stable organizations (Fig 2.14), although they do not necessarily have larger objective values.

The maximum and minimum of these discrete solutions provide a tight empirical bound to our discrete solutions from any single initialization, for example, the hierarchical initialization in Fig 2.10. Shown in Fig 2.13 the objective values for the continuous optima monotonically decrease with larger K, whereas those for the discrete optima gradually decrease by and large, but not monotonically. We also see that the hierarchical initialization helps to approach an optimal discrete solution most of the time. Examining these values with their corresponding segmentations, we find that ε itself is not very indicative for selecting the best K.

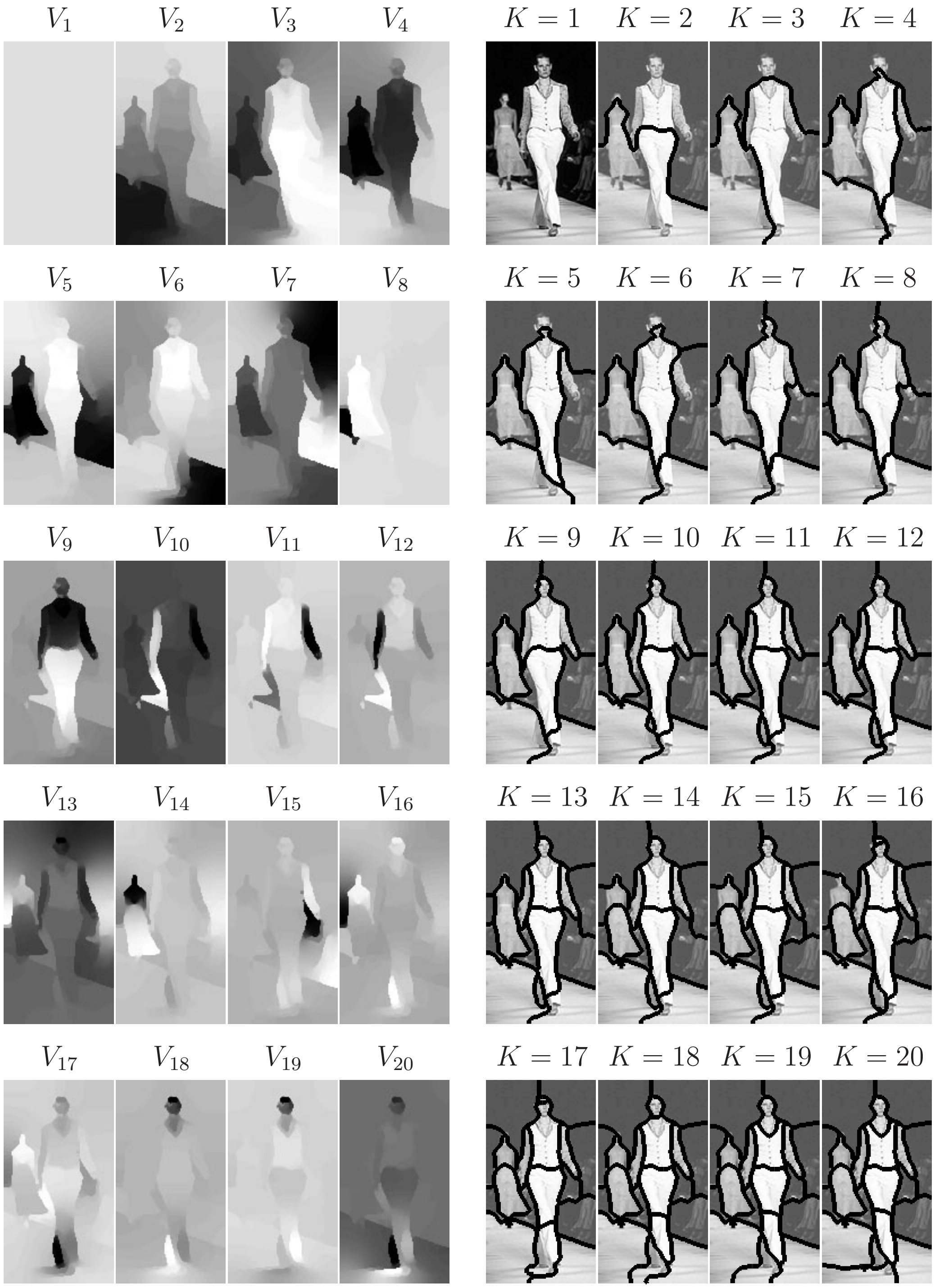

V_1 V_2 V_3 V_4
$K = 1$ $K = 2$ $K = 3$ $K = 4$
V_5 V_6 V_7 V_8
$K = 5$ $K = 6$ $K = 7$ $K = 8$
V_9 V_{10} V_{11} V_{12}
$K = 9$ $K = 10$ $K = 11$ $K = 12$
V_{13} V_{14} V_{15} V_{16}
$K = 13$ $K = 14$ $K = 15$ $K = 16$
V_{17} V_{18} V_{19} V_{20}
$K = 17$ $K = 18$ $K = 19$ $K = 20$

Figure 2.10: Multiclass spectral clustering for image segmentation. It takes 36 seconds to compute the 20 leading eigenvectors in MATLAB on a PC with 1GHz CPU and 1GB memory. Image size: 120×97. Each is a segmentation using the first K eigenvectors. The discretization process takes 0.1 up to 1.1 seconds.

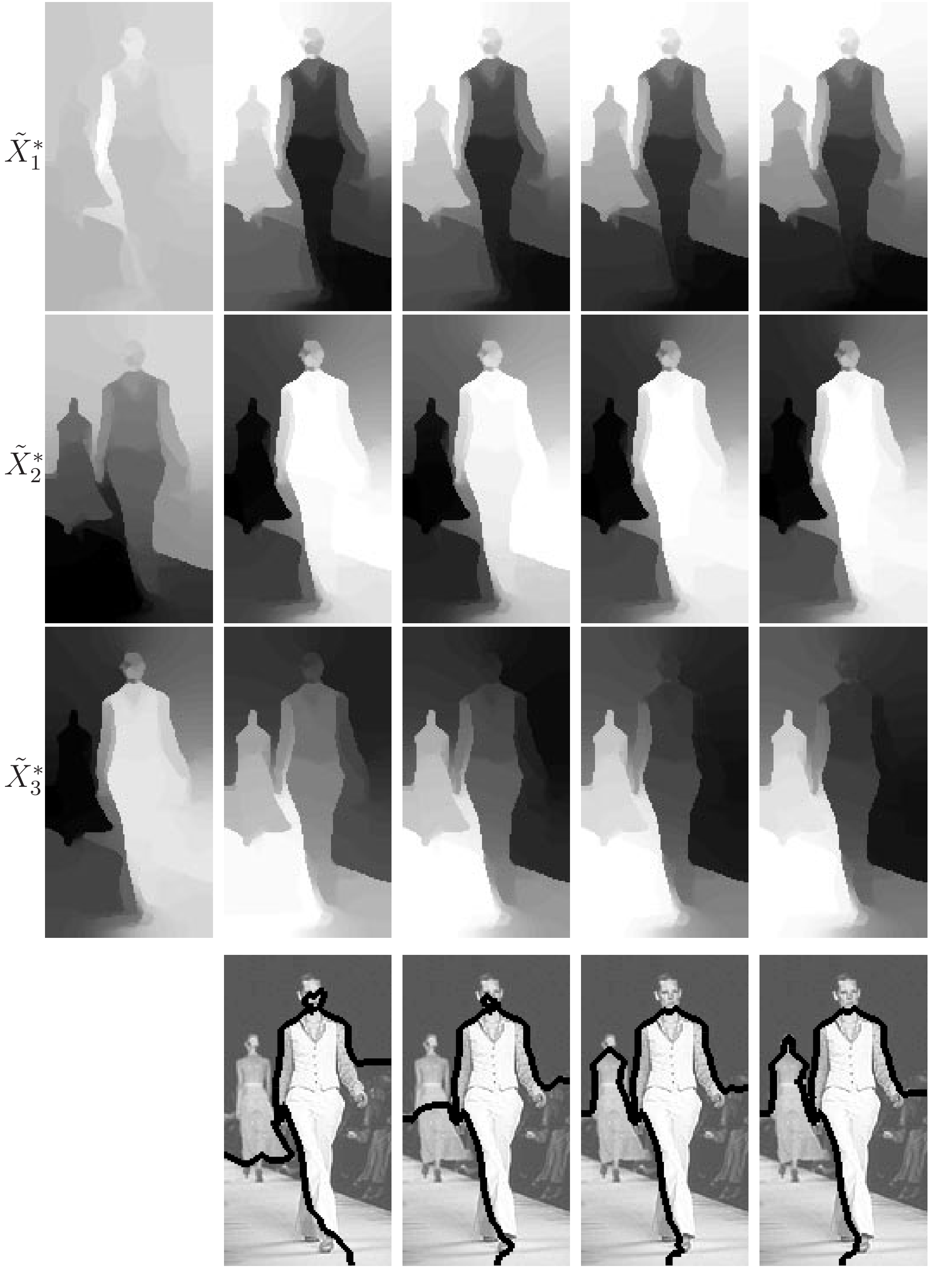


Figure 2.11: Progression of our algorithm for image segmentation. $K = 3$. Column #1 continuous optimum: $\tilde{X}^* = f^{-1}([V_1, V_2, V_3])$. Columns #2 till #5: discretization at iteration $t = 0, 1, 2, 3$: $\tilde{X}^* R^{*(t)}$ (rows #1-#3), $X^*(t)$ (row #4), with $\varepsilon(X^{*(t)})$ evolving from 0.9860, 0.9874, 0.9929 to 0.9932 upon convergence.

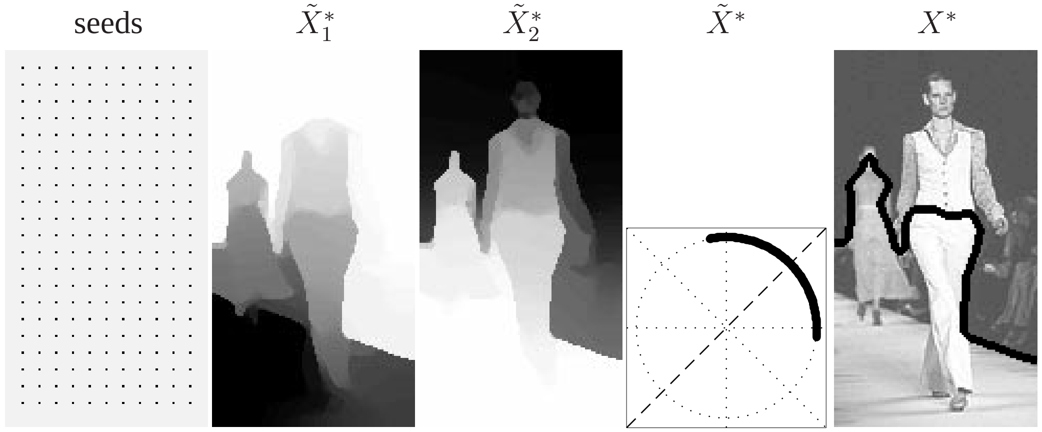
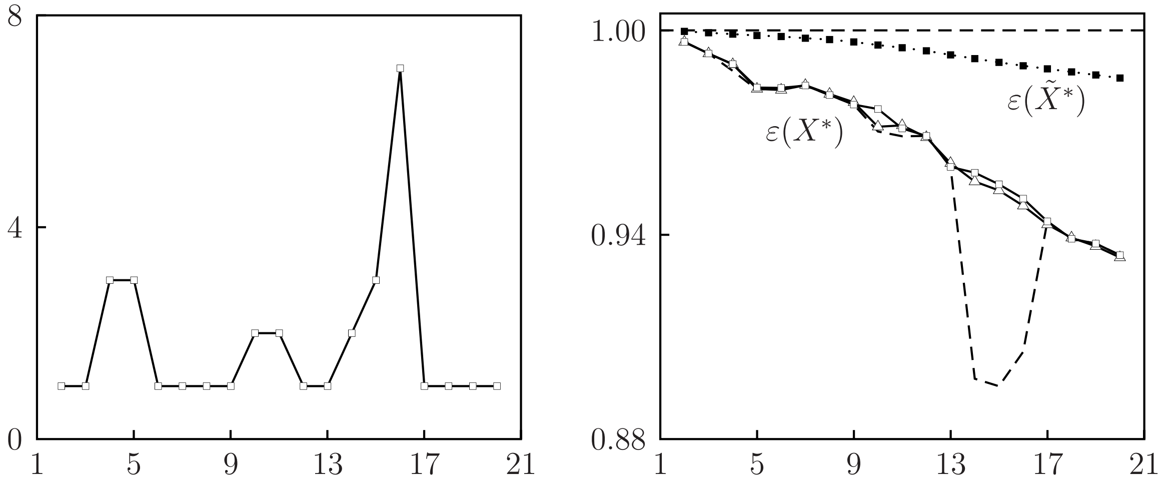


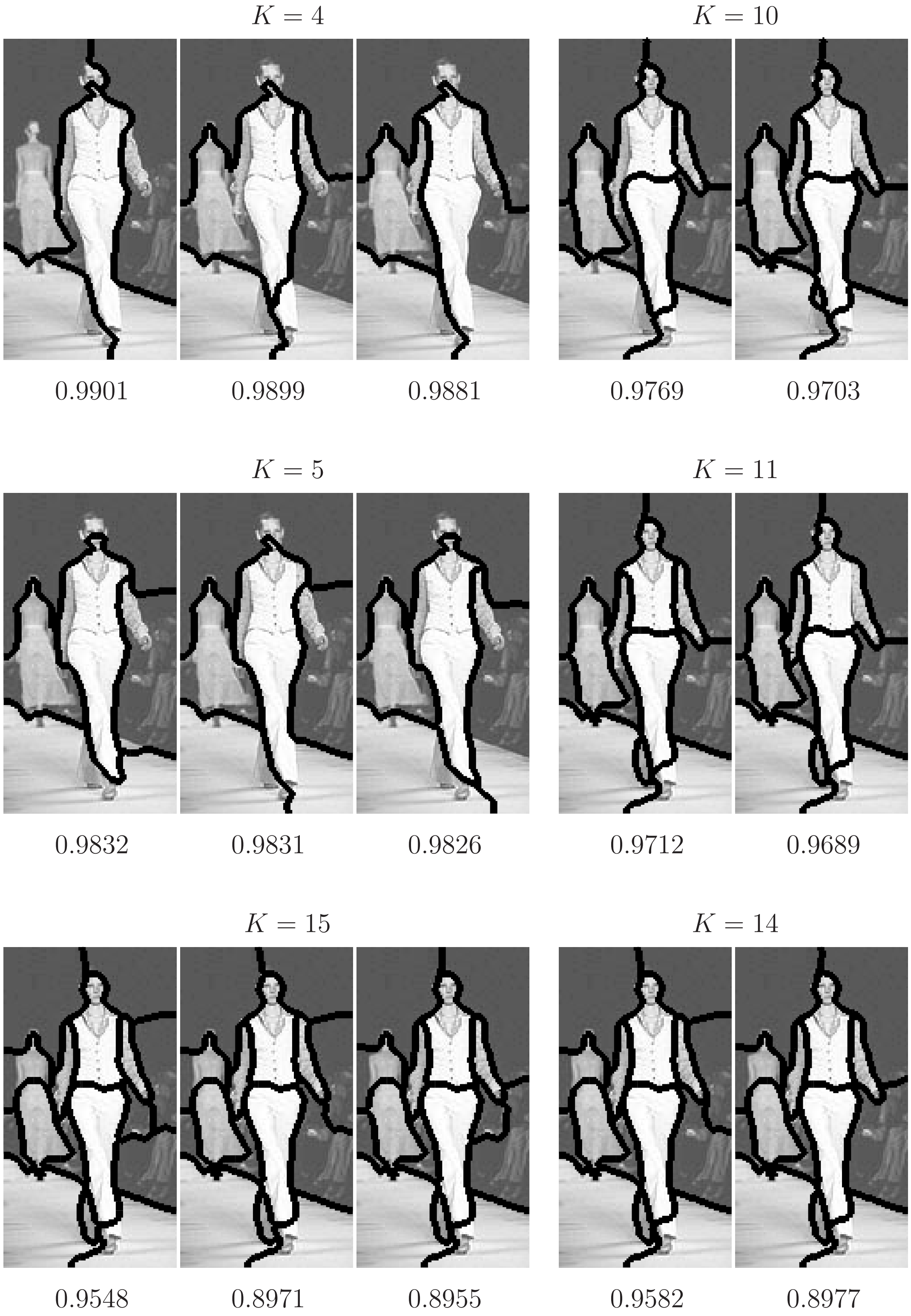
Figure 2.12: The optimal discrete solution for $K = 2$. All the pixel seeds that are used to initialize our discretization procedure lead to the same discrete optimal solution.



a: number of discrete optima

b: optimal objective values

Figure 2.13: Optimality of discrete solutions from our discretization procedure. The abscissa is K for both cases. a: number of discrete solutions. The discrete solution might depend on initialization. b: ε values for continuous optima (topmost, ■), best discrete solutions (second, □), worst discrete solutions (dashed line), and the discrete solutions from the hierarchical initializations (in-between △) in Fig 2.10.



$K = 4$

0.9901 0.9899 0.9881

$K = 10$

0.9769 0.9703

$K = 5$

0.9832 0.9831 0.9826

$K = 11$

0.9712 0.9689

$K = 15$

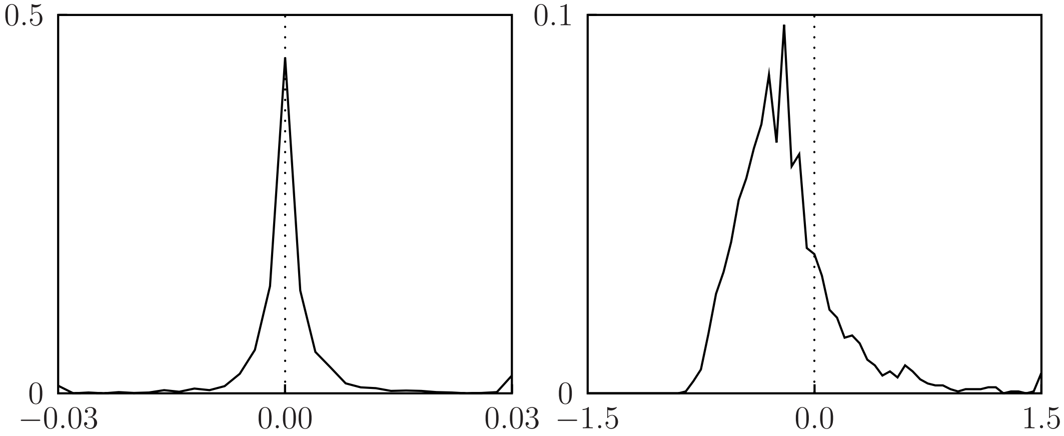
0.9548 0.8971 0.8955

$K = 14$

0.9582 0.8977

Figure 2.14: Multiple optimal discrete solutions. Numbers are their ε values.

Fig 2.15 shows that K-means on $\tilde{X}^*$ (Ng et al., 2002) can produce similar results but it may take twice as long to converge. In (Ng et al., 2002), a perturbation rationale is given for the need to normalize the eigenvectors, while the use of K-means is unjustified. K-means' similar results are a consequence of the continuous optima greatly reducing the chance for misclustering. Yet we observe that a good initial estimation is crucial for K-means, whereas our method is robust to a random initialization. This is not surprising because K-means introduces additional unwarranted assumptions, while our principled account has a clear criterion ϕ to optimize, which guarantees the near global optimality of discrete solutions under the orthogonal invariance of continuous optima.



a: difference in ε

b: difference in time

Figure 2.15: Performance comparison to K-means clustering on $\tilde{X}^*$. Both are estimated probability distribution of the relative difference between the two methods: $\frac{g-g_{Kmeans}}{g}$, where g is ε in a and running time in b. These statistics are collected over 100 Berkeley test images. Each image is segmented into 2 to 20 classes. Both codes are optimized to take advantage of the unit lengths of all data points, with the same initialization method.

We ran our algorithm on $450+$ real images. Fig 2.16 and Fig 2.17 are samples of our results on a set of fashion pictures and Berkeley test set. The number of classes K is manually chosen. We also tried various automatic selection methods. $\varepsilon''_K(X^*)$ seems to be most informative. However, this requires further research.

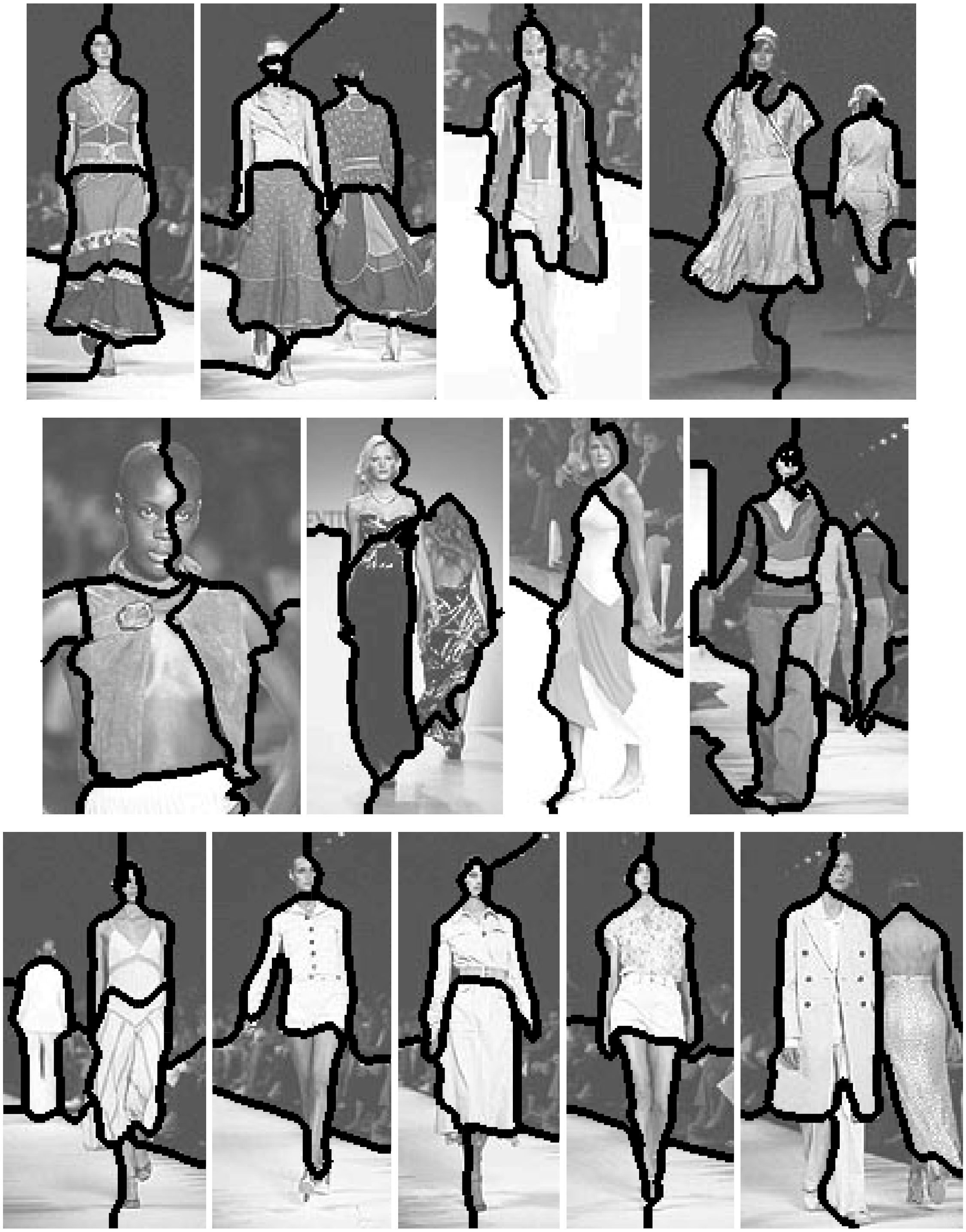


Figure 2.16: Multiclass segmentation on New York Spring 2002 fashion pictures.

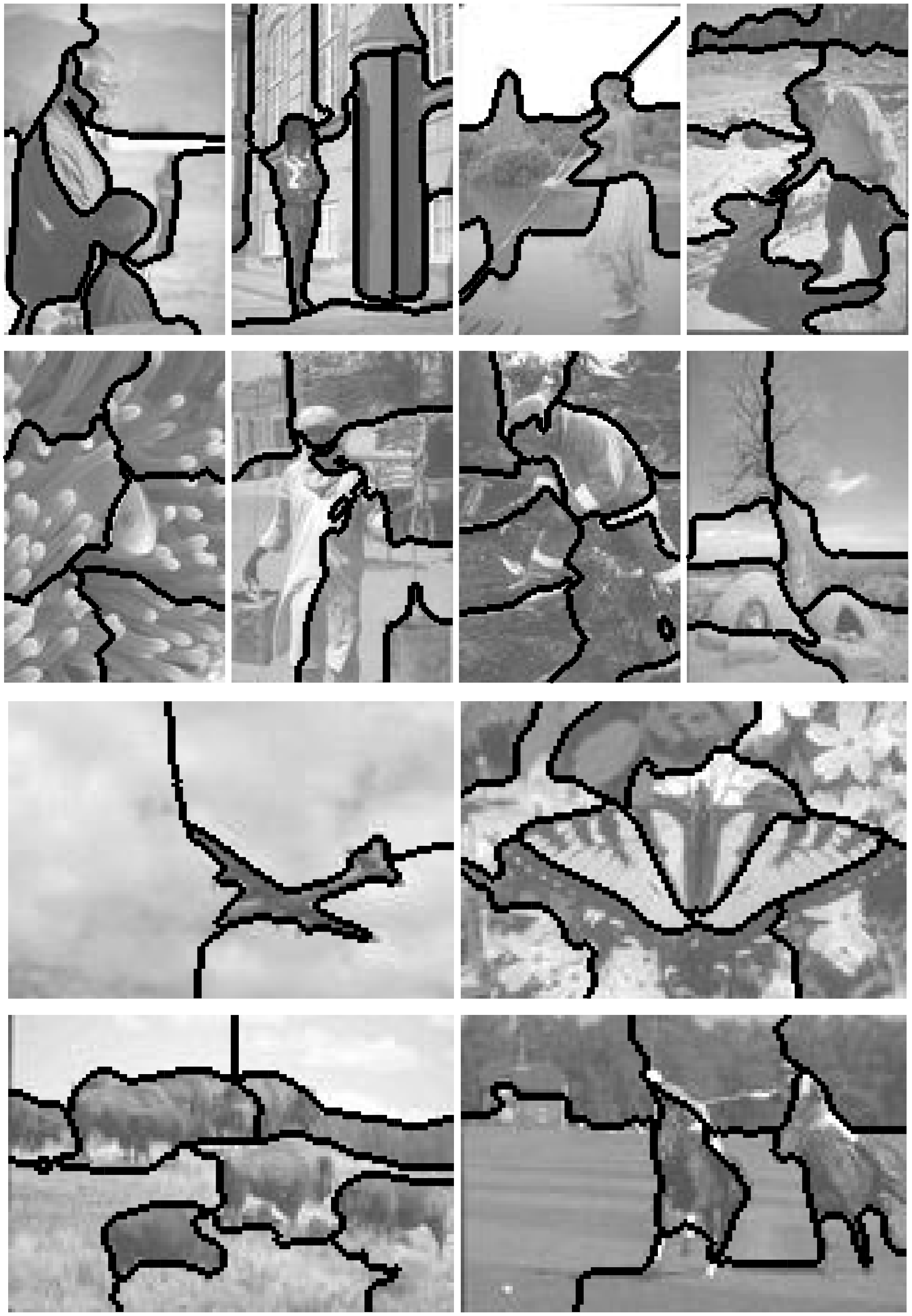


Figure 2.17: Multiclass segmentation on Berkeley test images.

2.4 Summary

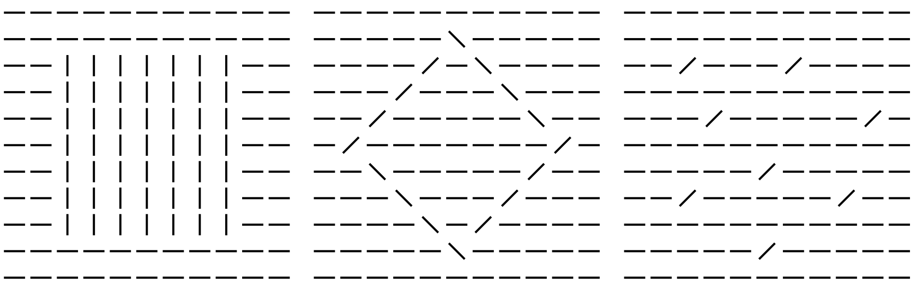
We gave a principled account of multiclass normalized cuts, starting from a relaxed continuous solution to the final discretization. It clarifies the role of eigenvectors as a generator for all continuous optima, based on which we developed a fast and flexible discretization procedure for finding a nearly global-optimal partitioning. We tested our method on real image segmentation and found results promising.

Our account also applies to various average cuts criteria since they only differ in the definitions of the normalization term (Shi and Malik, 2000), which is nonessential to all the developments here. We expect the use of our algorithm to improve those spectral methods as well.

Chapter 3

Repulsion and Regularization

Visual processing starts by extracting local features such as oriented edges. The features detected at an early stage are then grouped into meaningful entities such as regions, boundaries and surfaces. The goal of pre-attentive visual segmentation (Li, 2000) is to mark conspicuous image locations which most likely demand further processing. These locations not only include boundaries between regions, but also smooth contours and popout targets in a background (Fig 3.1).



a: boundary. b: contour. c: popout.

Figure 3.1: The goal of pre-attentive segmentation is to mark conspicuous image locations, which could be caused by a: region boundaries, b: smooth contours and c: popout targets. In these examples, the similarity of features within figure and ground are confounded with the dissimilarity between figure and ground.

It has long been assumed that regions are foremost characterized by features which are homogeneous within the areas. Their values are then compared in neighbourhoods to locate boundaries between regions (Li, 2000). This view of feature discrimination for grouping is supported by evidence in neurophysiology on elaborate feature detectors in visual cortex (van Essen, 1985), in psychophysics on visual search (Treisman, 1985) and in modeling on texture segmentation (Julesz, 1984; Bergen and Adelson, 1988; Malik and Perona, 1990). Some other approaches of texture segmentation go beyond the analysis of features obtained from image filters by also modeling the interactions between filters (Zhu et al., 1998). These Markov random field models (Geman and Geman, 1984) capture contextual dependencies and other statistical characteristics of texture features.

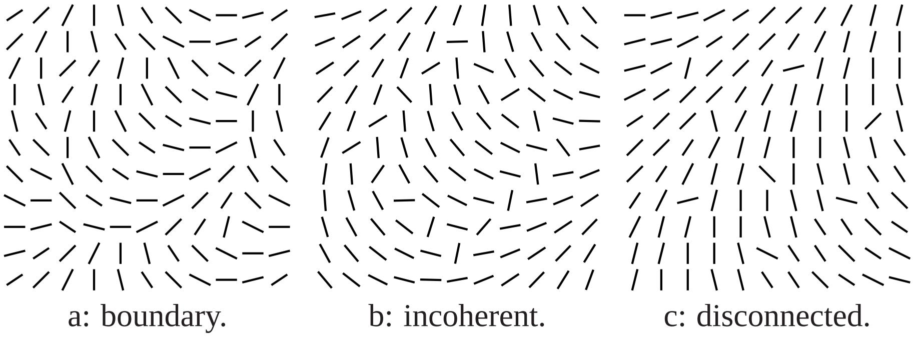

a: boundary. b: incoherent. c: disconnected.

Figure 3.2: Local feature contrast alone is sufficient to perceptually link dissimilar elements together. a: Boundary by local orientation contrast. b: Figure without curvilinearity. c: Spatially disconnected figure with low inter-element similarity.

However, it has been shown that when feature similarity within an area and feature differences between areas are teased apart, the two aspects of perceptual organization, *association* and *segregation*, can contribute somewhat independently to grouping (Beck, 1982; Julesz, 1986; Sagi and Julesz, 1987; Nothdurft, 1993). In particular, when feature values change continuously in areas, it is the local feature contrast, rather than the feature properties themselves, that is more important for the perceived grouping (Fig 3.2). These results have motivated a few models of pre-attentive vision which *directly* localize region boundaries through lateral interactions between edge detectors (Nothdurft, 1997; Li, 2000).

Such contextual feature analysis for grouping can be modeled in a graph-theoretic framework, where each element is denoted by a node and the relationships between the elements are described by weights attached to the edges connecting the nodes. For example, Gestalt grouping factors, such as proximity, similarity, continuity and symmetry, can be first evaluated through a comparison of feature values associated with the elements, then combined into a measure summarizing the overall grouping compatibility (Wu and Leahy, 1993; Shi and Malik, 1997; Puzicha et al., 1998; Gdalyahu et al., 1998; Sharon et al., 2000). While Gestalt laws have always stressed the similarity of elements in grouping, the effect of local feature contrast cannot be captured in a framework that only models similarity grouping. Fig 3.2c gives an example where completely dissimilar elements that are spatially disconnected can be perceived as a figure simply because they are locally dissimilar to a *common* background.

In this chapter, we integrate such dissimilarity cues into a conventional method that only groups by similarity. We generalize the normalized cuts criterion (Shi and Malik, 1997) to handle both types of cues. Using a simplified scenario, we derive necessary and sufficient conditions for our model to segregate figure from ground. This helps us to understand perceptual popout and its relationship to general grouping problems. We demonstrate these concepts on image segmentation.

3.1 Grouping with Attraction and Repulsion

In graph approaches for segmentation, an image is described by a weighted graph $\mathbb{G} = (\mathbb{V}, \mathbb{E})$, where each pixel becomes a node and a measure of *feature similarity* between two pixels is attached to the edge connecting their nodes. For an image of N pixels, all such pairwise comparisons are summarized in an $N \times N$ weight matrix W. W is assumed nonnegative and symmetric. To segment an image into K regions is to partition all nodes into K disjoint sets. We denote a K-way node partitioning as $\Gamma_{\mathbb{V}}^K = \{\mathbb{V}_1, \dots, \mathbb{V}_K\}$, where $\mathbb{V} = \cup_{l=1}^K \mathbb{V}_l$ and $\mathbb{V}_l \cap \mathbb{V}_k = \varnothing$, $l \neq k$.

There are many criteria for selecting a good graph partitioning. Here we will use the normalized cuts criterion (Shi and Malik, 1997). However, the concepts to be developed also apply to other criteria.

A brief account of normalized cuts is as follows. For node sets $\mathbb{P}$, $\mathbb{Q} \subset \mathbb{V}$, let $\mathrm{links}(\mathbb{P}, \mathbb{Q}; W)$ be the total connections from $\mathbb{P}$ to $\mathbb{Q}$; let $\mathrm{degree}(\mathbb{P}; W)$ be the total connections from $\mathbb{P}$ to all the nodes in the graph; let $\mathrm{linkratio}(\mathbb{P}, \mathbb{Q}; W)$ be the connection ratio from $\mathbb{P}$ to $\mathbb{Q}$:

$$\begin{aligned}
\mathrm{links}(\mathbb{P}, \mathbb{Q}; W) &= \sum_{p \in \mathbb{P}, q \in \mathbb{Q}} W(p, q), & (3.1)\\
\mathrm{degree}(\mathbb{P}; W) &= \mathrm{links}(\mathbb{P}, \mathbb{V}; W), & (3.2)\\
\mathrm{linkratio}(\mathbb{P}, \mathbb{Q}; W) &= \frac{\mathrm{links}(\mathbb{P}, \mathbb{Q}; W)}{\mathrm{degree}(\mathbb{P}; W)}. & (3.3)
\end{aligned}$$

In particular, $\mathrm{linkratio}(\mathbb{P}, \mathbb{P}; W) + \mathrm{linkratio}(\mathbb{P}, \mathbb{V} \setminus \mathbb{P}; W) = 1$. The so-called normalized associations and normalized cuts criteria are defined as:

$$\begin{aligned}
\mathrm{knassoc}(\Gamma_{\mathbb{V}}^{K}; W) &= \frac{1}{K} \sum_{l=1}^{K} \mathrm{linkratio}(\mathbb{V}_l, \mathbb{V}_l; W) & (3.4)\\
\mathrm{kncuts}(\Gamma_{\mathbb{V}}^{K}; W) &= \frac{1}{K} \sum_{l=1}^{K} \mathrm{linkratio}(\mathbb{V}_l, \mathbb{V} \setminus \mathbb{V}_l; W). & (3.5)
\end{aligned}$$

Since $\mathrm{knassoc}(\Gamma_{\mathbb{V}}^{K}; W) + \mathrm{kncuts}(\Gamma_{\mathbb{V}}^{K}; W) = 1$, maximizing the associations and minimizing the cuts are equivalent.

3.1.1 Representation

We use two *nonnegative* weight matrices, A and R, to describe respectively the feature similarity and dissimilarity between all pairs of nodes in the graph.

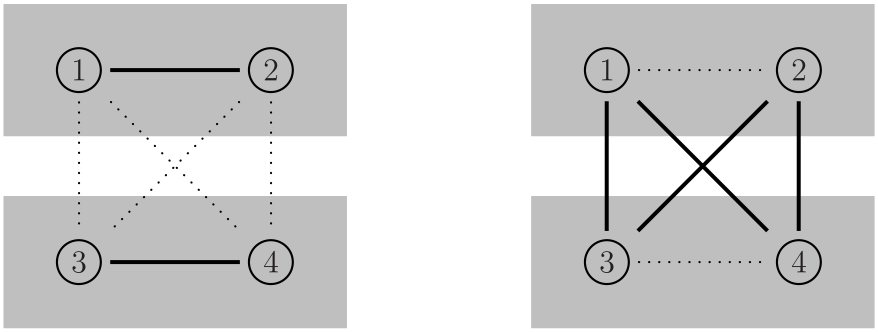
There might be some confusion at this point: why do we need these two measures which seem to imply each other? The fact is that these numbers encode not just the degree of similarity or dissimilarity, but also its confidence. Therefore, one measure alone inevitably has ambiguity. For example, there are two possible interpretations to weight $A(i, j) = 0$: either i and j are not similar at all, or we are not certain about their similarity.

Consider a concrete example in image segmentation. We choose $A(i, j)$ to be a Gaussian function of intensity difference between pixels i and j. Such comparisons would introduce erroneous grouping cues for pixels far away from each other. We thus restrict the evaluation to nearby pixels and set $A(i,j) = 0$ for all pairs of pixels beyond a certain distance. This zero value does not mean two pixels are not similar, but rather we don't know whether they are similar or not.

Therefore, A and R complement each other precisely because similarity measurements have different certainties. If the range of all measurements is between 0 and 1, then $A(i,j) = 0.1$ is not equivalent to $R(i,j) = 0.9$. The reason is that the former is associated with a very low confidence, while the latter is associated with a high confidence. For images or any other data, we often derive cues with different levels of confidence, some leaning toward an attraction nature, some toward a repulsion nature. Such a pair of representations is not redundant.

3.1.2 Criteria

Intuitively, attraction indicates association and repulsion indicates segregation. A good clustering maximizes within-group associations and between-group segregation, but minimizes their complements (Fig 3.3). Quantitatively,



a: association by attraction A b: segregation by repulsion R

Figure 3.3: Grouping criteria. $\Gamma_{\mathbb{V}}^2 = \{\{1,2\},\{3,4\}\}$. A good grouping maximizes connection ratios of thick-lined edge weights and minimizes those of dotted-lined weights. a: Large within-group associations and small between-group associations are desired. b: Large between-group segregation and small within-group segregation are desired.

$\text{linkratio}(\mathbb{P},\mathbb{P};A)$: within-group association

$\text{linkratio}(\mathbb{P},\mathbb{V}\setminus\mathbb{P};R)$: between-group segregation.

Therefore, for each of the K partitions, we combine attraction and repulsion by:

$$
\begin{aligned}
\text{knassoc}(\Gamma_{\mathbb{V}}^{K}) = \frac{1}{K}\sum_{l=1}^{K} \Bigg[& \\
& \text{linkratio}(\mathbb{V}_l, \mathbb{V}_l; A) \cdot \frac{\text{degree}(\mathbb{V}_l; A)}{\text{degree}(\mathbb{V}_l; A) + \text{degree}(\mathbb{V}_l; R)} + \\
& \text{linkratio}(\mathbb{V}_l, \mathbb{V} \setminus \mathbb{V}_l; R) \cdot \frac{\text{degree}(\mathbb{V}_l; R)}{\text{degree}(\mathbb{V}_l; A) + \text{degree}(\mathbb{V}_l; R)} \Bigg], \quad (3.6) \\
\text{kncuts}(\Gamma_{\mathbb{V}}^{K}) = \frac{1}{K}\sum_{l=1}^{K} \Bigg[& \\
& \text{linkratio}(\mathbb{V}_l, \mathbb{V} \setminus \mathbb{V}_l; A) \cdot \frac{\text{degree}(\mathbb{V}_l; A)}{\text{degree}(\mathbb{V}_l; A) + \text{degree}(\mathbb{V}_l; R)} + \\
& \text{linkratio}(\mathbb{V}_l, \mathbb{V}_l; R) \cdot \frac{\text{degree}(\mathbb{V}_l; R)}{\text{degree}(\mathbb{V}_l; A) + \text{degree}(\mathbb{V}_l; R)} \Bigg]. \quad (3.7)
\end{aligned}
$$

Again the duality is maintained since $\text{knassoc} + \text{kncuts} = 1$. Therefore the two goals are equivalent. We make no distinction further and denote $\varepsilon = \text{knassoc}$.

3.1.3 Computational Solution

We use partition matrix $X = [X_1, \ldots, X_K]$ to represent $\Gamma_{\mathbb{V}}^{K}$, where X_l is a binary membership indicator for group l. Since a pixel is only assigned to one partition, there is an exclusion constraint on X: $X\,1_K = 1_N$, where 1_d denotes the $d \times 1$ vector of all 1's.

We define the equivalent weight matrix $\hat{W}$ and equivalent degree matrix $\hat{D}$:

$$
\begin{aligned}
\hat{W} &= A - R + D_R, && (3.8) \\
\hat{D} &= D_A + D_R, && (3.9)
\end{aligned}
$$

where $D_W = \text{Diag}(W 1_N)$ is the degree matrix of W. We use Diag to denote a diagonal matrix made from its vector argument and diag to denote a column vector made from the diagonal of its matrix argument.

We introduce scaled partition matrix Z, where $Z = X(X^T \hat{D} X)^{-\frac{1}{2}}$. It naturally satisfies $Z^T \hat{D} Z = I$, where I is an identity matrix.

With these symbols, we rewrite the normalized cuts criterion as:

$$\text{maximize} \quad \varepsilon(\Gamma_{\mathbb{V}}^K) = \frac{1}{K} \sum_{l=1}^{K} \frac{X_l^T \hat{W} X_l}{X_l^T \hat{D} X_l^T} = \frac{1}{K} \operatorname{tr}(Z^T \hat{W} Z), \tag{3.10}$$

$$\text{subject to} \quad Z^T \hat{D} Z = I \tag{3.11}$$

$$X \in \{0, 1\}^{N \times K}, \quad X\, 1_K = 1_N. \tag{3.12}$$

This program is in the same form as conventional normalized cuts, thus the same computational procedure applies. That is, we first ignore the discrete constraints on X and obtain an optimal solution of Z by solving a generalized eigenvalue problem. Let (V, S) be the eigenvectors of $(\hat{W}, \hat{D})$, where $S = \operatorname{Diag}(s)$ has nonincreasingly ordered eigenvalues. Based on Gershgorin's theorem, we have $|s_l| \leq 2, \forall l$. All maximizers of ε in the relaxed continuous domain are generated by the eigenvectors corresponding to the K largest eigenvalues:

$$\max_Z \varepsilon(Z) = \varepsilon([V_1, \ldots, V_K]) = \frac{1}{K} \sum_{l=1}^{K} s_l \tag{3.13}$$

We then transform these optima to the space of X by a normalization procedure. Next we solve a discretization problem, where a discrete solution satisfying Eqn (5.6) yet closest to the continuous optima is sought (Yu and Shi, 2003).

We examine two extreme cases. When there is no repulsion, $\hat{W} = A$ and $\hat{D} = D_A$. This case is reduced to the conventional normalized cuts (Shi and Malik, 1997), where 1_N is the eigenvector of $(\hat{W}, \hat{D})$ with the largest eigenvalue of 1. suggesting that all nodes are one group. When there is no attraction, $\hat{W} = D_R - R$ and $D = D_R$. 1_N is the eigenvector of $(\hat{W}, \hat{D})$ with an eigenvalue of 0. When we have both attraction and repulsion, 1_N is no longer an eigenvector. Indeed, attraction tends to bind elements together, while repulsion tends to break elements apart. The optimal partitioning results from the balance of these two factors. We also see that the often considered trivial eigenvector 1_N is coincidental. It only happens when the weight matrix is entirely attraction.

3.2 Negative Weights and Regularization

Given a symmetrical weight matrix W, let

$$W = W_+ - W_- = A - R, \tag{3.14}$$

where W_+ and W_- contain the absolute values of all positive and negative entries of W respectively. We regard W_+ as attraction A and W_- as repulsion R, and interpret normalized cuts on A and R as that on W. The equivalent eigensystem $(\hat{W}, \hat{D})$ is thus

$$\begin{aligned} \hat{W} &= W + D_{W_-}, & (3.15) \\ \hat{D} &= D_{W_+} + D_{W_-}. & (3.16) \end{aligned}$$

With the introduction of repulsion, we no longer require weight matrices to be nonnegative for graph partitioning. Furthermore, Eqn (3.14) is not the only way to decompose a symmetric matrix into attraction and repulsion. In general, with an arbitrary nonnegative matrix Δ, we have:

$$W = (W_+ + \Delta) - (W_- + \Delta) = A - R. \tag{3.17}$$

The previous case corresponds to $\Delta = 0$. If we interpret W using $A = W_+ + \Delta$ and $R = W_- + \Delta$, the continuous optimum is then given by the eigenvectors of

$$(\hat{W} + D_\Delta,\ \hat{D} + 2D_\Delta). \tag{3.18}$$

We see that, no matter how much variation the entries of Δ take, the only effect Δ has on the solution is through D_Δ. We choose $D_\Delta = \delta \cdot I$, where δ is a scalar.

This extra degree of freedom provides us with a means to regularize the solutions of spectral methods. When $\hat{D}$ has near-zero values for some nodes, the segmentation by the eigenvectors of $(\hat{W}, \hat{D})$ is numerically unstable. This situation occurs when a coherent figure is embedded in a random background. In the attraction case, this problem can be remedied by the addition of a small constant baseline connection weight. However, such a technique lacks any theoretical justification and alters the measurements of pairwise affinity. In our current framework, we can introduce a baseline connection to both the attraction component

W_+ and the repulsion component W_-, so that their effects are canceled out while the stability of grouping is improved.

In other words, with the dual representation, we can encode the value of similarity and the value of confidence independently. Adding a non-zero Δ to W_+ and W_- amounts to increasing the confidence without changing the value of similarity measures, so that we have more certainty about the grouping for nodes with near-zero connections. A regularized solution thus reveals more stable groups.

How does the grouping change with the amount of regularization? This question is related to the limiting behavior of the eigensolution at $\delta = \infty$.

Theorem 2 (Regularization at Limit). *Let $\Delta = \delta I$, $\delta > 0$. Let (V^δ, S^δ) be the eigenvectors and eigenvalues of the normalized weight matrix P^δ, where*

$$P^\delta = (\hat{D} + 2D_\Delta)^{-1}(\hat{W} + D_\Delta), \tag{3.19}$$

Let (U, Λ) be the eigendecomposition of $P^0 - \mathrm{Diag}(\mathrm{diag}(P^0))$. Then we have:

$$\begin{aligned} (V^\delta, S^\delta) &\approx (U, \Lambda + 0.5\, I), \quad \delta \gg \max \hat{D}. && (3.20)\\ V^\infty &= \textit{any } N \times N \textit{ orthogonal matrix} && (3.21)\\ S^\infty &= 0.5\, I. && (3.22) \end{aligned}$$

Proof. From Eqn (3.19), V^δ and S^δ are equivalently the generalized eigenvectors and eigenvalues of $(\hat{W} + D_\Delta,\ \hat{D} + 2D_\Delta)$ since $(\hat{W} + D_\Delta)V^\delta = (\hat{D} + 2D_\Delta)V^\delta S^\delta$, i.e. they are the optimal normalized cuts solutions in the continuous domain. When $\delta \gg \max \hat{D}$, we have:

$$P^\delta(i,i) = \frac{\hat{W}(i,i) + \delta}{\hat{D}(i,i) + 2\delta} = \frac{1}{2} \cdot \frac{\hat{W}(i,i) + \delta}{0.5\hat{D}(i,i) + \delta} \approx \frac{1}{2} \tag{3.23}$$

$$P^\delta(i,j) = \frac{\hat{W}(i,j)}{\hat{D}(i,i) + 2\delta} = P^0(i,j) \cdot \frac{\hat{D}(i,i)}{\hat{D}(i,i) + 2\delta} \approx P^0(i,j) \cdot \frac{1}{2\delta},\ j \neq i \tag{3.24}$$

Therefore, when δ is sufficiently large,

$$P^\delta - 0.5\, I \approx \left[P^0 - \mathrm{Diag}(\mathrm{diag}(P^0)) \right] \cdot \frac{1}{2\delta}, \tag{3.25}$$

i.e. $P^\delta - 0.5\ I$ becomes a scaled version of a constant matrix. Since subtracting a scaled identity matrix ($0.5I$) does not change the eigenvectors (but shifts eigenvalues by the same scale constant), the eigensolution of P^δ remains as $(U, \Lambda + 0.5\ I)$ until $\frac{1}{2\delta}$ becomes vanishingly small so that $P^\delta - 0.5\ I = 0$. In other words,

$$P^\infty = 0.5\ I. \tag{3.26}$$

The conclusions result. □

We can consider δ sufficiently large when $\delta = 10 \max D$. If μ is the minimum floating number of a digital representation, then when $\delta > \frac{1}{2\mu}$, $P^\delta \approx P^\infty$. For $\delta \in [10 \max D, \frac{1}{2\mu}]$, (V^δ, S^δ) stays the same. In short, regularization admits a wide range of δ, a fact that will be verified in the experiments section.

3.3 Algorithm with Repulsion and Regularization

Given weight matrix W or pair (A, R), and regularization parameter δ:

1. Compute degree matrices for attraction and repulsion:

$$W = A - R = W_+ - W_-$$
$$D_A = \text{Diag}(A1),\ D_R = \text{Diag}(R1).$$

2. Compute the equivalent weight and degree matrices with regularization:

$$\hat{W} = W + D_R + \delta \cdot I$$
$$\hat{D} = D_A + D_R + 2\delta \cdot I.$$

3. Compute the K largest eigenvectors of $(\hat{W}, \hat{D})$.
4. Compute a discrete partition matrix X from the eigenvectors.

3.4 Understanding Popout

We study the simple 4-node graph in Fig 3.3. Let

$$W = \begin{bmatrix} 0 & x & y & y \\ x & 0 & y & y \\ y & y & 0 & z \\ y & y & z & 0 \end{bmatrix}, \qquad X = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{3.27}$$

where x, y, z denote figure-to-figure, figure-to-ground, ground-to-ground connections respectively (Amir and Lindenbaum, 1998b; Amir and Lindenbaum, 1998c). Each column of X gives a 2-class partitioning. Due to symmetry, we only need to consider the four cases in X. We determine the conditions on x, y and z so that the partitioning $\Gamma_{\mathbb{V}}^2 = \{\{1,2\},\{3,4\}\}$ is guaranteed.

	z	y	x
$\delta = 0$	1	$(-\infty,\ 0)$	$(1 - y - \sqrt{1 - 2y + 9y^2},\ +\infty)$
	1	$[0,\ 1]$	$(\dfrac{2y^2}{1+y},\ +\infty)$
	1	$(1,\ +\infty)$	$(-y + 2y^2,\ +\infty)$
	-1	$(-\infty,\ -1)$	$(\dfrac{-2y^2}{1-y},\ \dfrac{-1+2y+8y^2}{2})$
	-1	$[-1,\ -\frac{1}{2}]$	$(-y - 2y^2,\ \dfrac{-1+2y+8y^2}{2})$
$\delta = \infty$	1	$(-\infty,\ 0)$	$(-1 + 2y,\ +\infty)$
	1	$[0,\ 1]$	$(\max(0, \dfrac{-1+8y}{7}),\ +\infty)$
	1	$(1,\ +\infty)$	$(-7 + 8y,\ +\infty)$
	-1	$(-\infty,\ -1)$	$(\dfrac{1+8y}{7},\ +\infty)$
	-1	$[-1,\ -\frac{7}{8}]$	$(7 + 8y,\ +\infty)$
	-1	$[0,\ +\infty)$	$(1 + 2y,\ +\infty)$

Table 3.1: Feasible sets of parameters for Eqn (3.27) and Fig 3.3.

Since scaling W does not change the grouping, we assume $z = 1$. By requiring ε for the first column of X to be larger than that for any other column, after a lengthy derivation, we obtain the feasible sets of x and y. They are given in Table 3.1 and plotted in Fig 3.4.

In Fig 3.4, repulsion and regularization greatly expand the regions of affinity values that lead to the desired grouping. These effects are summarized in Table 3.2. With negative figure-ground connections arising from figures defined by local feature contrast, repulsion allows a figure with weak between-element similarity to pop out. With negative ground-ground connections arising from fragmented background, regularization allows a coherent foreground to stand out.

The problem of fragmented background has led (Perona and Freeman, 1998) to adopt an unbalanced criterion which emphasizes the coherence within the figure but not the ground. However, an unbalanced criterion tends to favor small local clusters and thus miss global grouping structures. Here we show that the same goal can be achieved with a balanced criterion in the attraction-repulsion framework.

3.5 Experiments

We use a Mexican hat function of feature difference to calculate both attraction and repulsion between data points. It is implemented as the difference of Gaussians:

$$h(x; \sigma_1, \sigma_2) = G(x, \sigma_1) - G(x, \sigma_2) \tag{3.28}$$

$$G(x, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp^{-\frac{1}{2}(\frac{x}{\sigma})^2} \tag{3.29}$$

$$h(r_0; \sigma_1, \sigma_2) = 0, \qquad r_0(\sigma_1, \sigma_2) = \sigma_1 \sqrt{\frac{2\ln(\frac{\sigma_1}{\sigma_2})}{(\frac{\sigma_1}{\sigma_2})^2 - 1}} \tag{3.30}$$

$$h(r_-; \sigma_1, \sigma_2) = \min_x h(x; \sigma_1, \sigma_2), \qquad r_-(\sigma_1, \sigma_2) = \sqrt{3} \cdot r_0(\sigma_1, \sigma_2). \tag{3.31}$$

$h(x)$ is attraction if positive, repulsion if negative and neutral if zero. There are two critical points, r_0, where the affinity changes from attraction to repulsion; and

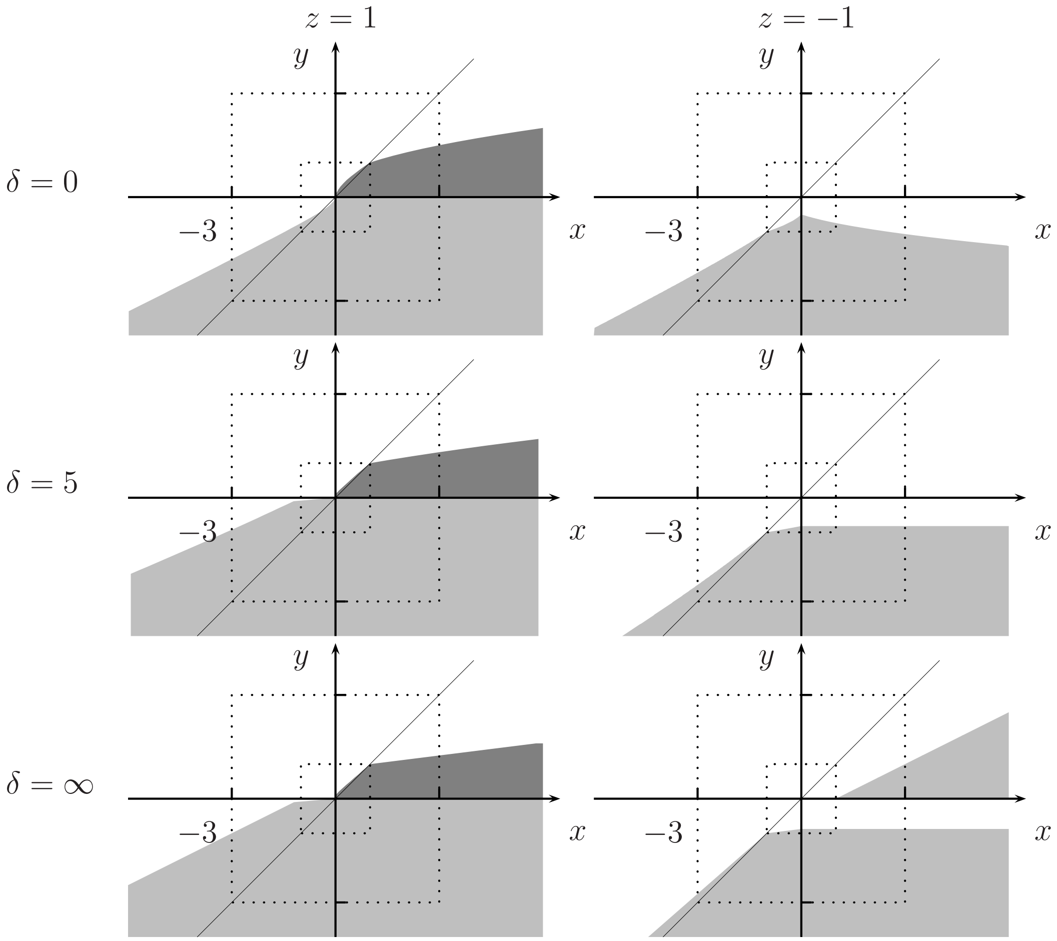


Figure 3.4: Repulsion and regularization help figure-ground segregation. Here x, y and z are figure-figure, figure-ground and ground-ground affinity. The shaded areas indicate feasible regions for figure-ground segregation. The darker areas are those with attraction alone. When $z = 1$, the ground is made of similar elements. When $y > 0$, x has to increase rapidly (quadratic). However, if $y < 0$, x can be even more repulsive than y. Therefore, with attraction, only coherent figures pop out, while with repulsion, even incoherent figures can pop out. When $z = -1$, the ground is incoherent. If y is attraction, no coherent figure ($x > 0$) can be segmented. If y is repulsion, then a figure pops out even if $x < y$. With regularization, measured by δ, the solution space in general expands. In particular, a sufficiently coherent figure (with a linear $x - y$ relationship) can pop out from a random ground,which would be otherwise impossible.

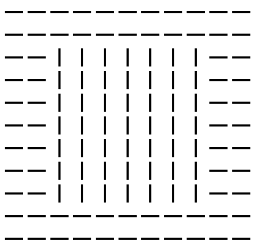
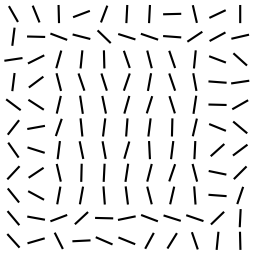
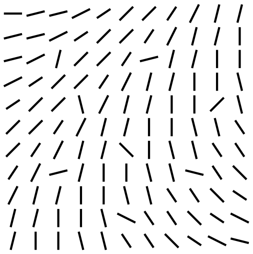
figure \ ground	coherent	incoherent
coherent	 attraction	 regularization
incoherent	 repulsion	

Table 3.2: Perceptual popout illustrates distinct major contributions of attraction, repulsion and regularization. Attraction is most effective for detecting a coherent figure against a coherent ground. With repulsion, dissimilar elements pop out against a common ground. With regularization, a coherent figure pops out from a random ground.

r_-, where the affinity is the largest repulsion (Fig 3.5).

For synthetic images, we compute affinity for pairs of pixels within a city-block distance r. We first illustrate the effects of repulsion on a toy example, where x is the intensity difference of two pixels. Fig 3.6 shows that repulsion not only binds heterogeneous objects, but also requires fewer local feature compar-

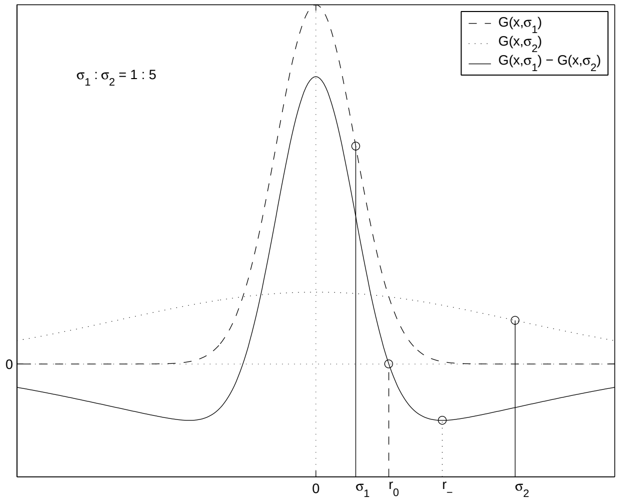


Figure 3.5: Calculate pairwise attraction and repulsion using Mexican hat functions based on difference of Gaussian. If x denotes the distance between two data points, then there is zero affinity (neither attraction nor repulsion) at $x = r_0$, maximum repulsion at r_-, and maximum attraction at 0.

isons. For attraction, since zero could mean either two pixels are highly dissimilar or they are not neighbours, the result with $r = 1$ has graded values over the background and the larger object. With a larger r, zero attraction is disambiguated and both objects come out as different groups until $r = 7$. If the objects have opposite contrasts, attraction cannot possibly group them, whereas repulsion – capturing local feature contrast – readily unites them against a common ground.

Fig 3.7 shows grouping results on bar configurations, where the feature we use is the orientation of bars. Attraction is good at grouping similar elements, but poor at detecting salient outlier groups. When the ground is coherent, repulsion between figure and ground can greatly reduce the pressure on figural coherence. Dissimilar elements pop out as one group from the background.

Fig 3.8 shows that neither attraction nor repulsion can segregate a coherent figure from a random ground. Under such circumstances, weights are highly unbalanced: figural nodes have large connections, while background nodes have nearly zero connections. A slight advantage in the weights of a few background

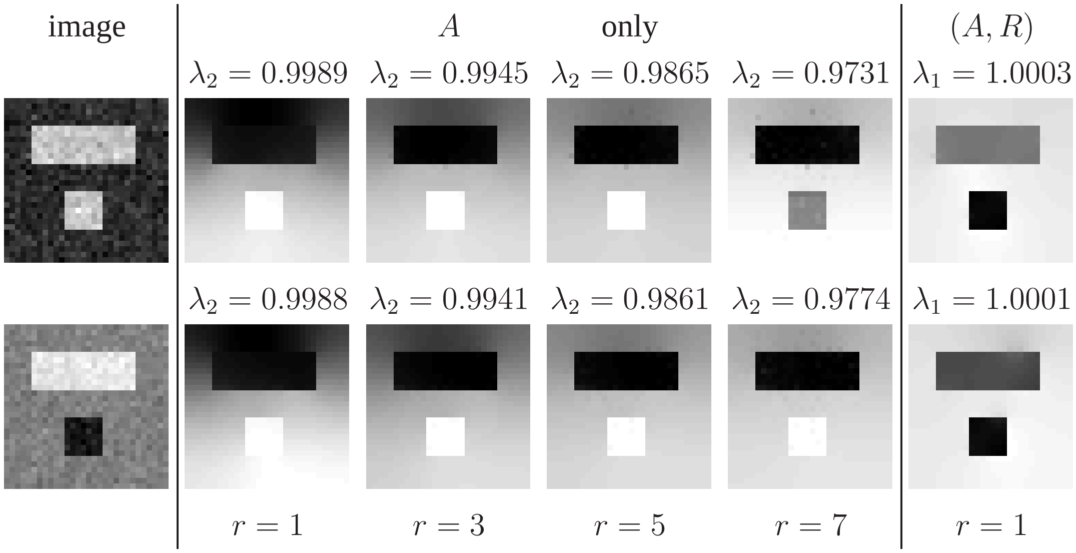


Figure 3.6: Repulsion can bind multiple objects with less computation. We use $\sigma_1 = 0.1$ to evaluate affinity by intensity, 3% of which become negative with $\beta = 5$. Column #2-5: results with attraction measured by $G(x; \sigma_1)$ with increasing r's. Column #6: result with affinity measured by $h(x; \sigma_1, 5\sigma_1)$. Row #1: the image has two rectangles of equal average intensity 0.8 against background of 0.5, added by Gaussian noise with standard deviation 0.03. A much larger neighborhood size is needed for attraction to achieve a comparable result with repulsion. Row #2: the smaller object now has an average intensity of 0.2. Even with a large r, the two objects cannot be united by attraction.

nodes would lead to one particular partitioning. Regularization of the weights, however, can improve the resistance to such perturbation, by increasing the degrees of the nodes without changing the relative sizes of the weights. As a result, only the relatively stable figure-ground organization could emerge.

Fig 3.8 also shows that the eigensolution changes little when δ increases from 10 to 10^{14}. It finally breaks down at $\delta = 10^{15}$. Since the maximum degree of all the nodes is about 2 and the floating point relative accuracy in our MATLAB is 2.2×10^{16}, these results corroborate the claim made in Theorem 2.

For real images, pixel affinity A is evaluated using the Gaussian function (standard deviation σ_e) of the largest magnitude of edges crossing the line connecting two pixels (Malik et al., 2001). This computation is restricted to all pixels within

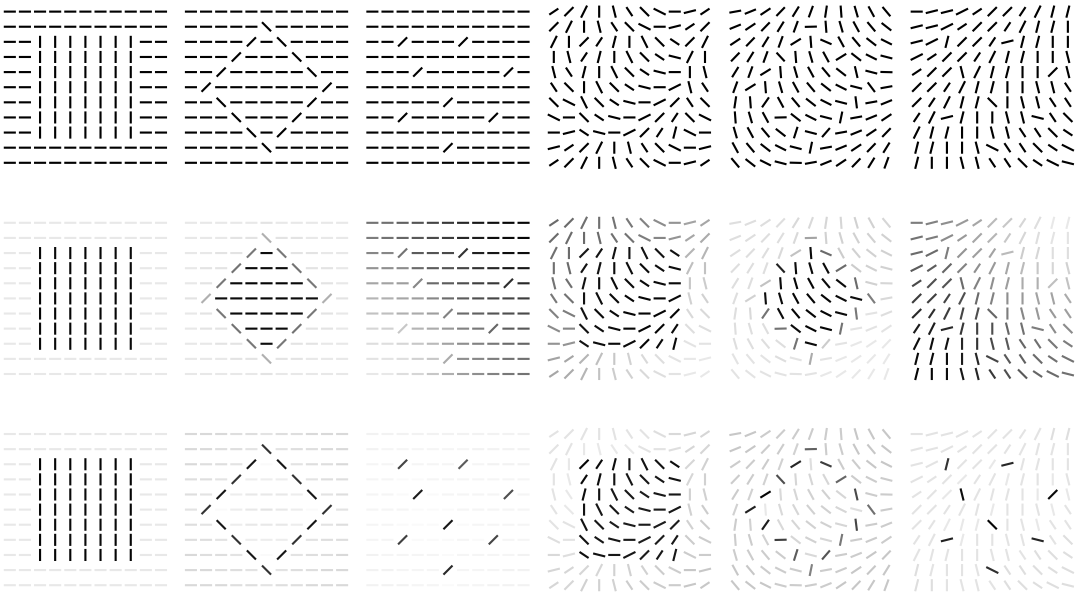


Figure 3.7: Pre-attentive segmentation on line segments. Row #1: stimuli. Row #2: eigenvector results by attraction. Row #3: eigenvector results with repulsion. $\sigma_1 = 15°$, $\sigma_2 = 45°$ are used for orientation. $r = 3$.

a distance of r. We derive repulsion by offsetting these values with a constant β:

$$W = A - \beta \cdot (A > 0), \tag{3.32}$$

Now small affinity of A becomes repulsion in W, which happens for pixels across strong edges, either at region boundaries or in a textured area.

A comparison of attraction and repulsion is given in Fig 3.9 to Fig 3.13. For all real images, we set $\sigma_e = 0.05$, $\beta = 0.1$, $\delta = 2.5$. A very small neighbourhood radius, $r = 2$, is used in calculating the pixel affinity. Otherwise, our simple formula to derive repulsion introduces too many wrong grouping cues. For attraction alone to achieve good results, a larger radius shall be used. However, even with a larger radius, as we have already seen in Fig 3.6, attraction would not be able to group small regions interspersed in a background.

The emergence of new grouping patterns with increasing repulsion and regularization is illustrated in Fig 3.14. Most evident in the phaseplots, where each

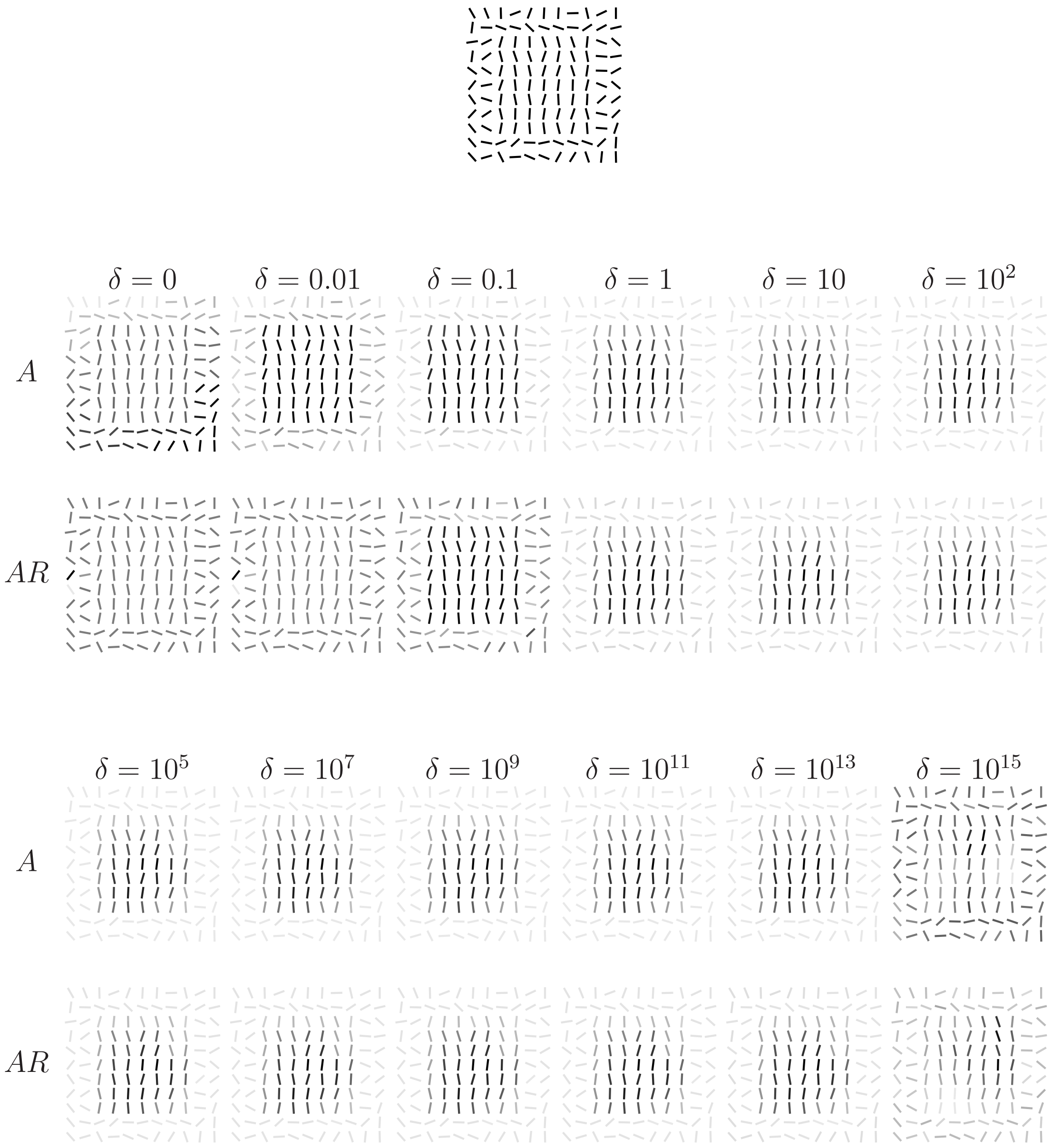


Figure 3.8: Regularization helps a coherent figure to pop out from a random background. Row #1: stimulus. Row #2,4: eigenvectors by attraction. Row #3,5: those with repulsion. Across the columns varies the amount of regularization δ. $\sigma_1 = 15^\circ$, $\sigma_2 = 45^\circ$ are used for orientation. $r = 1$.

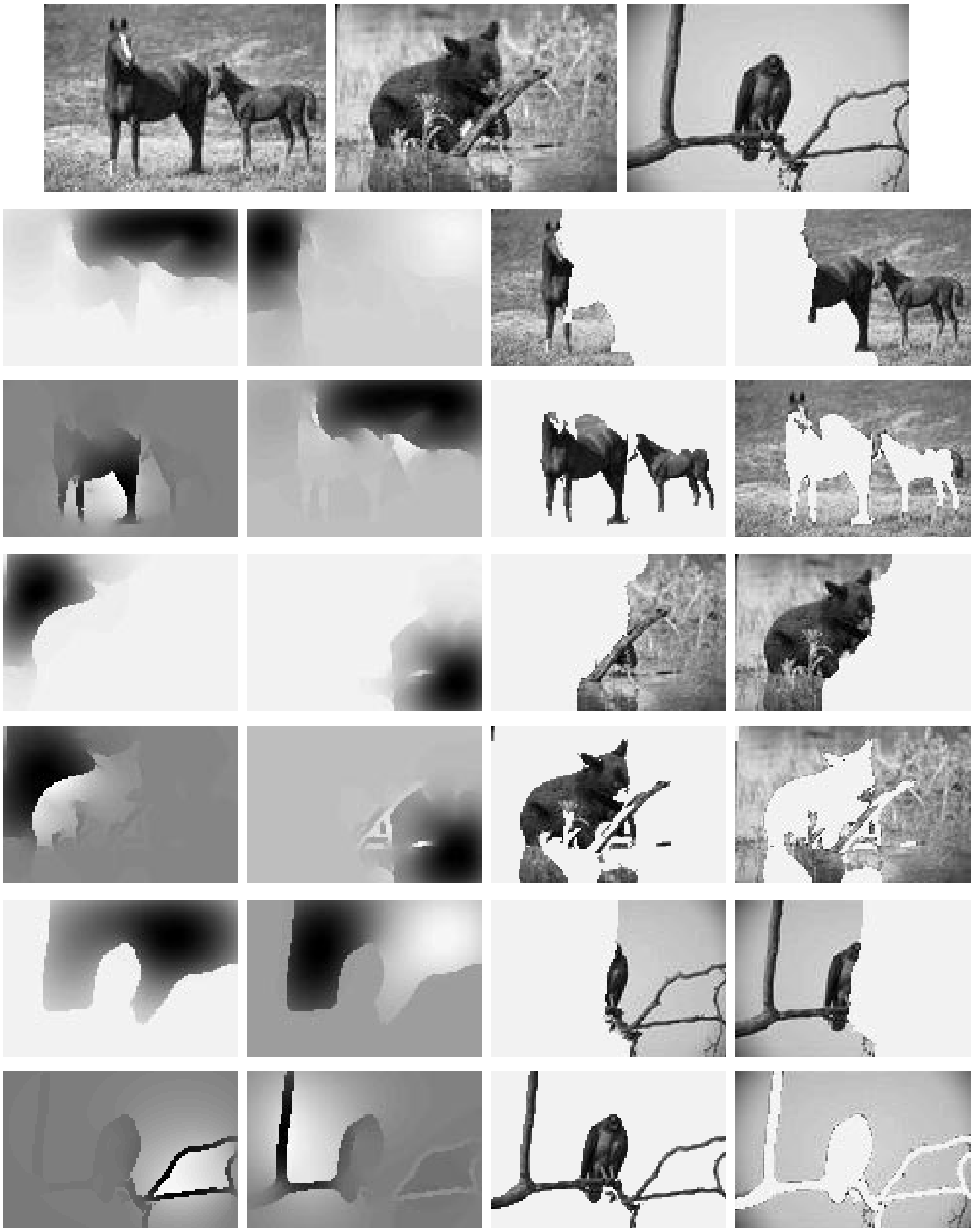


Figure 3.9: Segmentation with attraction and repulsion. Row #1: image size 120×80. Row #2,4,6: results with attraction. Row #3,5,7: results with repulsion. Columns #1,2: first two eigenvectors. Columns #3,4: two-class spectral segmentation based on the two eigenvectors shown in the same row.

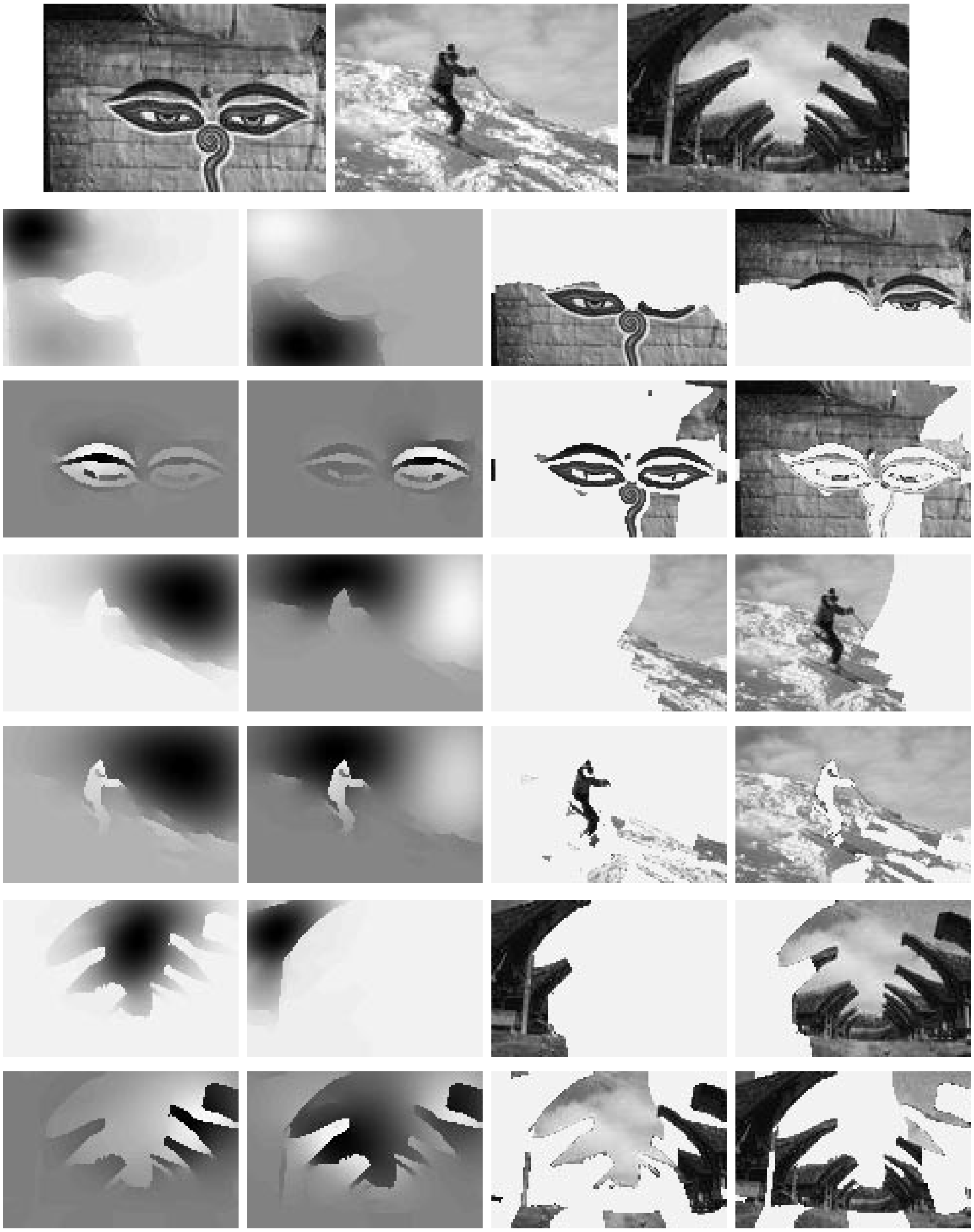


Figure 3.10: More results on 2-class spectral segmentation. Same convention as Fig 3.9.

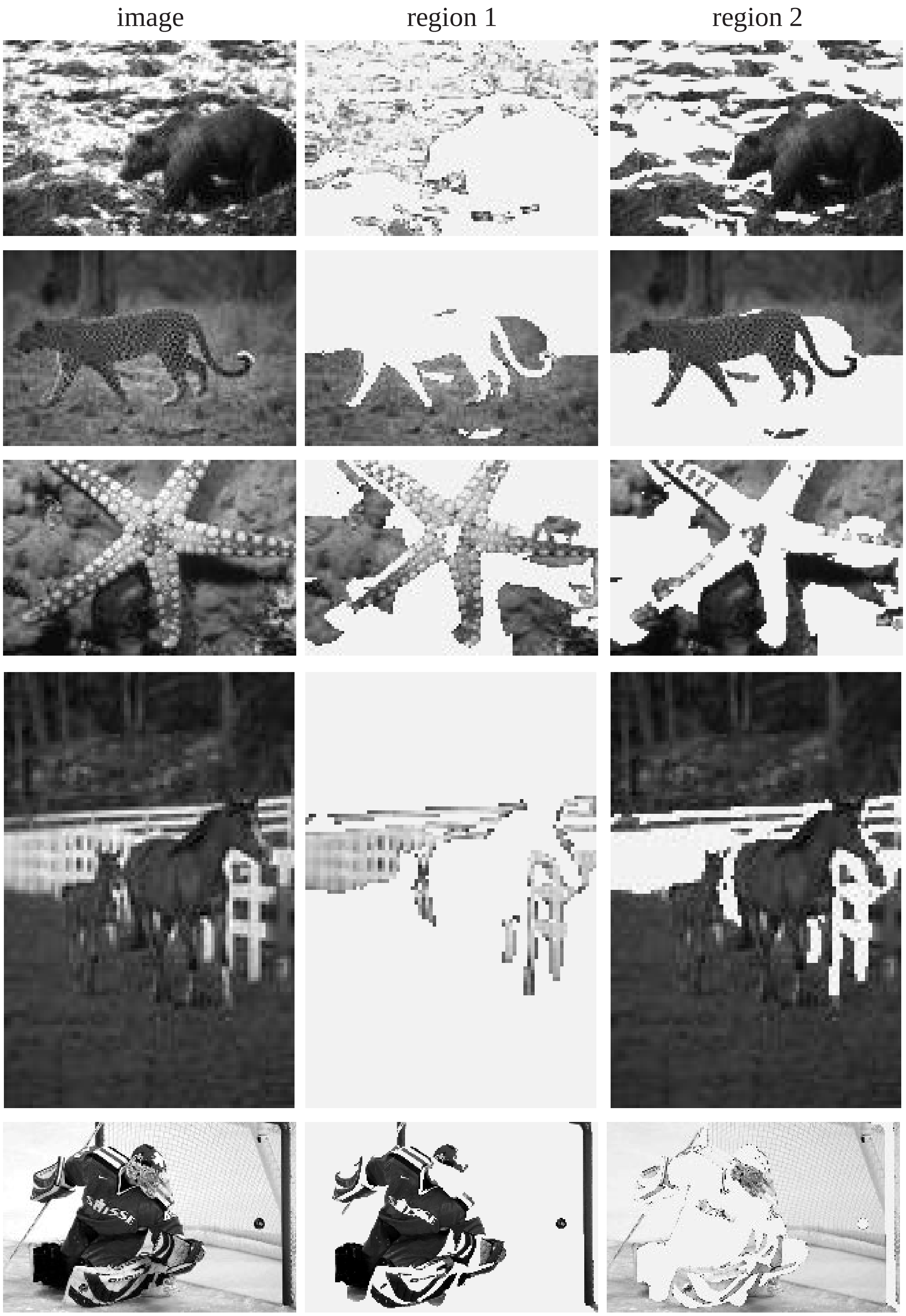


Figure 3.11: 2-class spectral segmentation with both attraction and repulsion.

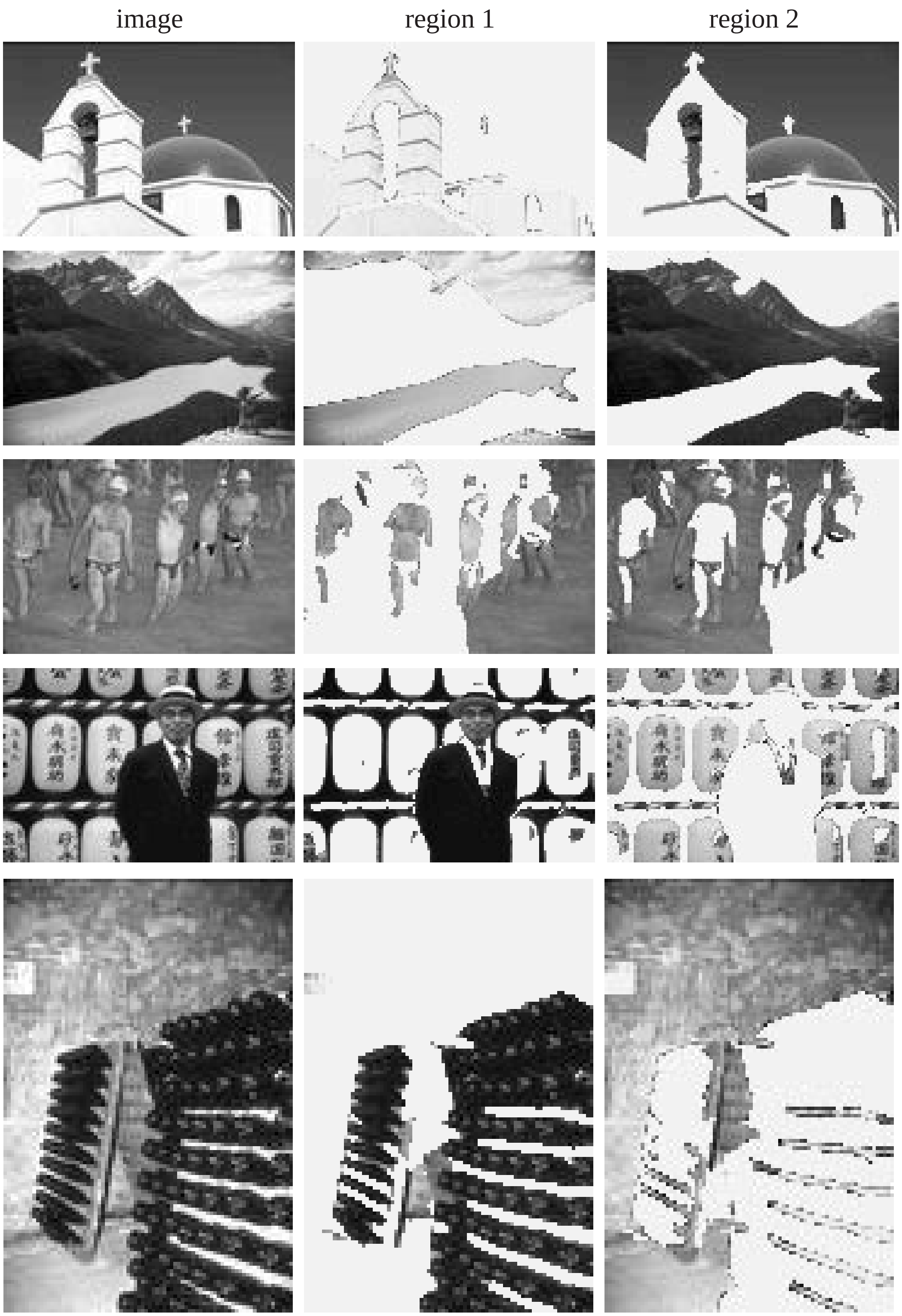


Figure 3.12: 2-class spectral segmentation with both attraction and repulsion.

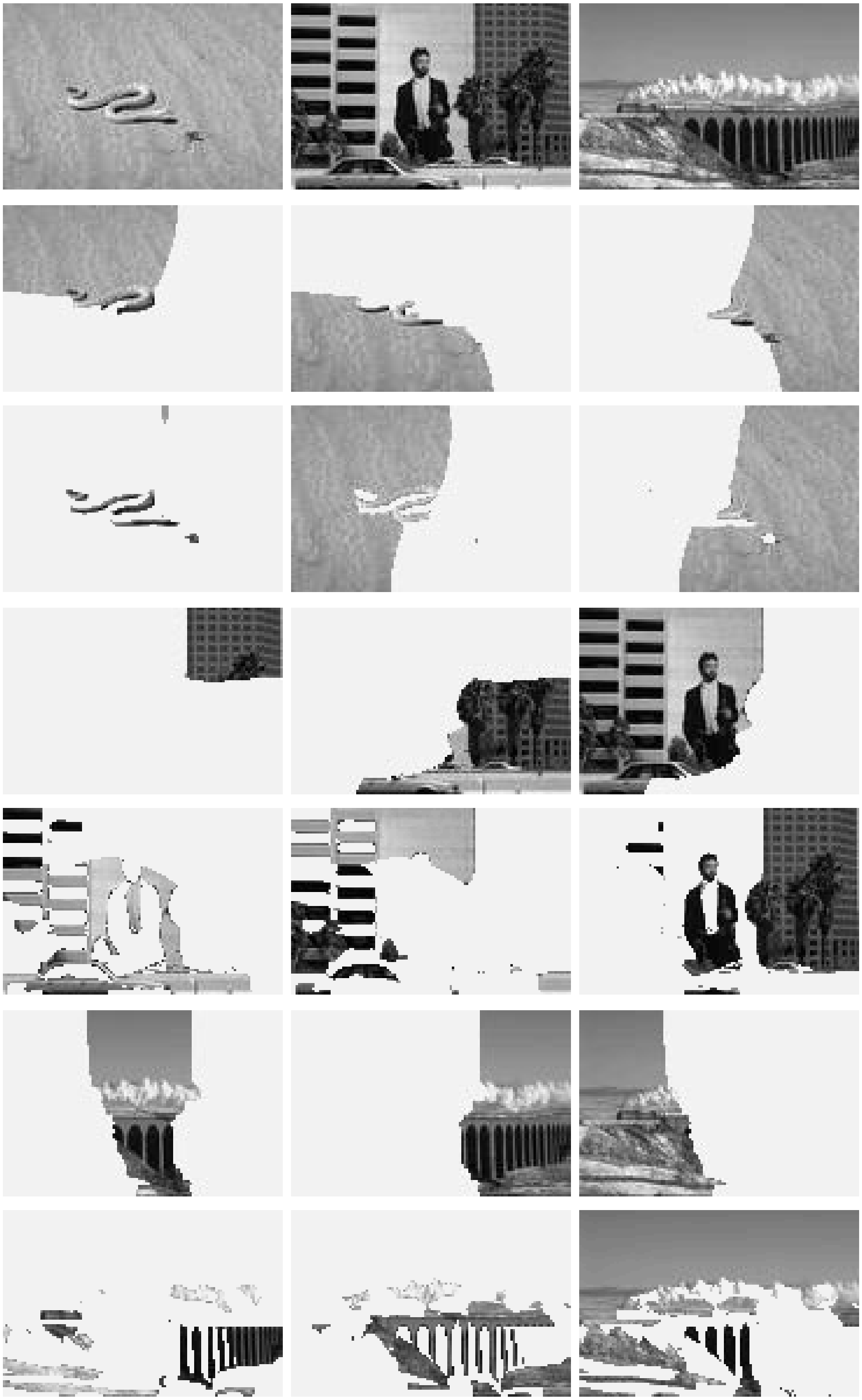


Figure 3.13: 3-class segmentation. Same convention as Fig 3.9 but no eigenvectors.

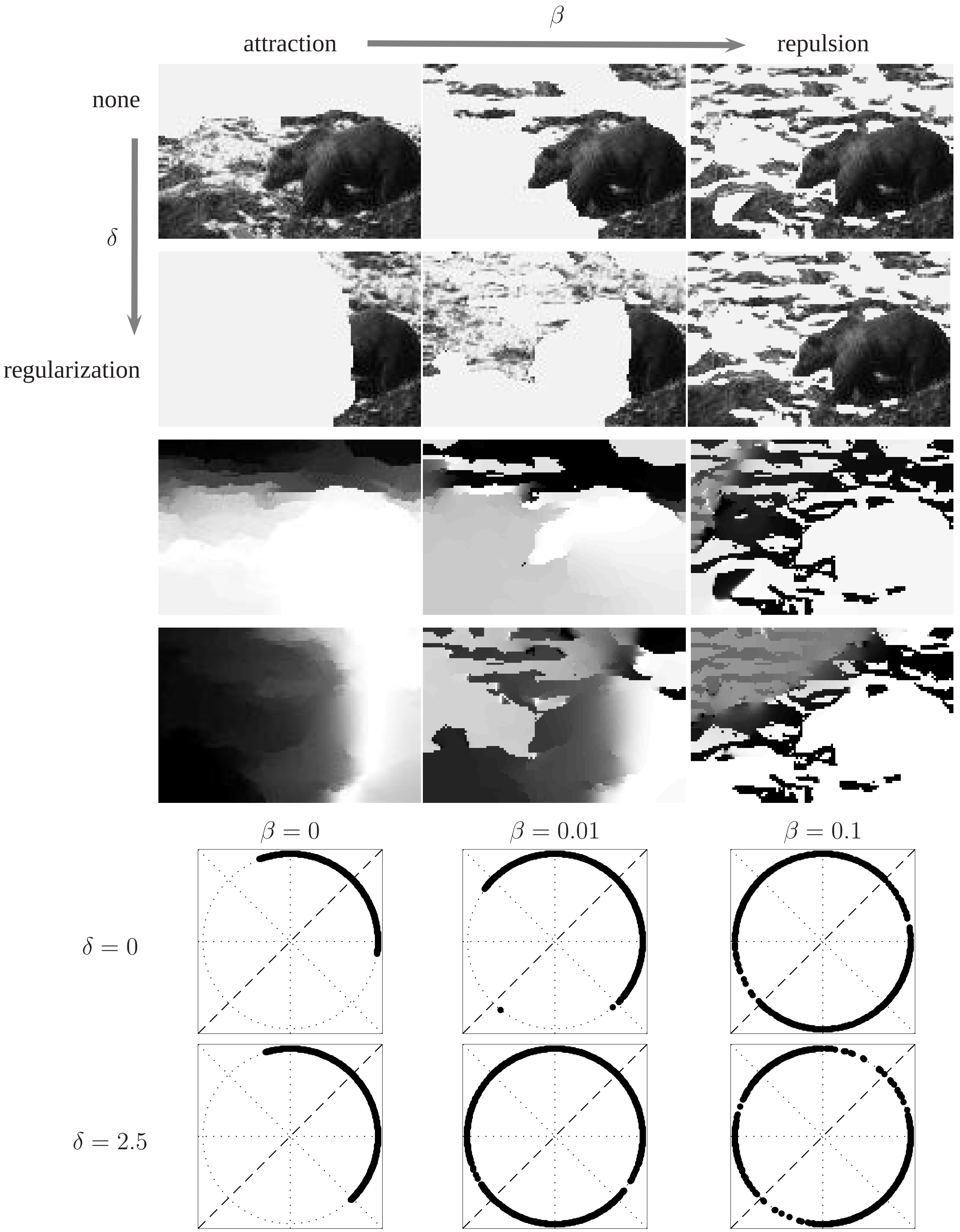


Figure 3.14: Refine segmentation with repulsion and regularization. The amount of repulsion β and regularization δ are varied in segmenting the first image in Fig 3.12. The results are presented in three sets of two rows: 1) one image region, 2) the continuous partitioning for this region, 3) 2-class continuous solution in phaseplots.

axis corresponds to one of the 2-class partitions in the continuous domain, repulsion pushes pixels into a wider range of values and regularization consolidates those values into chunks.

From these examples, we see that the inclusion of repulsion to affinity helps sharpen boundaries in segmentation and bring disconnected regions together. It greatly enriches the set of plausible partitions with only a few eigenvectors. For certain types of images, especially those comprised of small parts across a coherent background, the addition of repulsion effectively accentuates salient structures over a large region. The use of regularization helps discover stable groups in a segmentation, which is especially important for segmenting images with rich textures based on edge features.

3.6 Summary

We developed a grouping method unifying dual procedures of association by attraction and segregation by repulsion. Within this framework, we provided a theoretical basis for regularizing solutions of spectral graph partitioning algorithms.

We showed that all popout phenomena can be modeled with a balanced criterion, with attraction capturing feature similarity, repulsion capturing local feature contrast and regularization improving grouping stability.

We expanded graph partitioning to weight matrices with negative values, which provide a representation for negative correlations in constraint satisfaction problems. Efficient solutions to such formulations are thus possible.

Chapter 4

Grouping with Depth Orders

Figure-ground organization is a central problem in perception and cognition. It consists of two major processes: 1) depth segregation - the segmentation and ordering of surfaces in depth and assignment of *border ownerships* to relatively proximal objects in a scene (Kanizsa, 1979; Nakayama and Shimojo, 1992; Nakayama et al., 1989); 2) figural selection - the extraction and selection of a figure among a number of "distractors" in the scene. Evidence for both of these processes have been found in the early visual cortex (Knierim and van Essen, 1992; Lamme, 1995; Lee et al., 1998; Zhou et al., 2000; Dobbins et al., 1998).

In computer vision, figure-ground segregation is closely related to image segmentation and has been studied from both contour processing and region processing perspectives. Contour-based approaches perform contour completion by using good curve continuation (Grossberg and Mingolla, 1985; Heitger and von der Heydt, 1993; Mumford, 1993; Ullman, 1976; Williams and Jacobs, 1997), whereas region-based approaches perform image partitioning by using surface properties (Nitzberg et al., 1993; Shi and Malik, 1997; Zhu and Yuille, 1996). The formation of a global depth percept from local occlusion cues and the computation of layer organizations has also been modeled as an optimization process with a surface diffusion mechanism (Geiger and Kumaran, 1996; Geiger et al., 1998; Madarasmi et al., 1994; Yu et al., 2001).

It has long been recognized that there are strong connections between grouping and figure-ground processes. For example, they can be both derived from luminance, motion, continuation and symmetry (Palmer, 1999); closure in grouping is closely related to convexity, occlusion, and surroundedness in figure-ground: when a pair of symmetrical lines are grouped together, it essentially implies that the region between the contours is the figure and the surrounding area is the background.

However, depth segregation is not well integrated with image segmentation in computer vision. Either they are dealt with at separate processing stages (Blake and Zisserman, 1987; Wildes, 1991), or they are unified in a general formulation (Belhumeur, 1996) that has too many parameters to afford a tractable computation.

The difficulty of integrating figure-ground cues in a general grouping framework lies in the different natures of grouping cues. While grouping mainly looks at the association by feature similarity, figure-ground emphasizes the segregation by feature dissimilarity, and this dissimilarity could sometimes be directional. For example, local occlusion cues suggest some pixels to be in front of the others.

In this chapter, we develop a partitioning method in spectral graph theory that incorporates ordering cues. We propose a representation in which all possible pairwise relationships are characterized in two types of directed graphs, each encoding positive and negative correlations between data points. We generalize the normalized cuts criterion (Shi and Malik, 1997) to handle directional grouping cues. We show that the global-optima in the continuous domain can be obtained by solving generalized eigenvectors of Hermitian matrices in the complex domain. The real and imaginary parts of Hermitian matrices encode reciprocal and ordering relationships respectively. The phase angle separation defined by the eigenvectors in the complex plane determines the partitioning of data points, and the relative phase advance indicates the ordering of partitions.

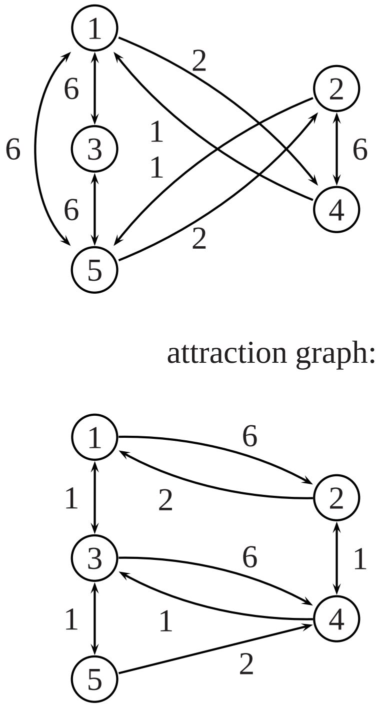
4.1 Grouping on Directed Graphs

Compared to the reciprocal similarity cues on intensity, color, texture and motion, occlusion cues encapsulate two distinct attributes: *repulsion* and *asymmetry*.

To integrate depth segregation with image segmentation, we need to generalize a grouping method in two ways: one is the inclusion of repulsion and directional cues; the other is a criterion for an ordered partitioning.

4.1.1 Representation of Asymmetric Relationships

We use two directed graphs to encode pairwise attraction and repulsion relationships: $\mathbb{G} = \{\mathbb{G}_A, \mathbb{G}_R\}$, $\mathbb{G}_A = (\mathbb{V}, \mathbb{E}_A, A)$, $\mathbb{G}_R = (\mathbb{V}, \mathbb{E}_R, R)$, where $\mathbb{V}$ is the set of all nodes to be grouped; $\mathbb{E}$ is the set of all edges defining pairwise relationships between nodes; A and R contain weights attached to these edges. Both A and R are *nonnegative*, but they can be asymmetric. See an example in Fig 4.1.



$$\begin{bmatrix} 0 & 0 & 6 & 2 & 6 \\ 0 & 0 & 0 & 6 & 1 \\ 6 & 0 & 0 & 0 & 6 \\ 1 & 6 & 0 & 0 & 0 \\ 6 & 2 & 6 & 0 & 0 \end{bmatrix}$$

attraction graph: $\mathbb{G}_A = (\mathbb{V}, \mathbb{E}_A, A)$

$$\begin{bmatrix} 0 & 6 & 1 & 0 & 0 \\ 2 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 6 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 & 0 \end{bmatrix}$$

repulsion graph: $\mathbb{G}_R = (\mathbb{V}, \mathbb{E}_R, R)$

Figure 4.1: Directed graph representation with nonnegative asymmetric weights.

Whereas directed repulsion can capture the asymmetry of relative depth orders between figure and ground, directed attraction might describe general compatibility between two pixels. For example, a pixel with reliable features is more likely to attract a pixel with ambiguous features, but not the other way around.

4.1.2 Criteria for an Ordered Partitioning

We recall a few definitions related to graph cuts (Yu and Shi, 2003). The links between node sets $\mathbb{P}, \mathbb{Q} \subset \mathbb{V}$ are the total connections from $\mathbb{P}$ to $\mathbb{Q}$; the degree of a set is its total connections to all the nodes; and the $\mathrm{linkratio}(\mathbb{P}, \mathbb{Q})$ is the proportion of the connections from $\mathbb{P}$ to $\mathbb{Q}$ over all those $\mathbb{P}$ has:

$$\begin{aligned}
\mathrm{links}(\mathbb{P}, \mathbb{Q}; W) &= \sum_{p \in \mathbb{P}, q \in \mathbb{Q}} W(p, q) && (4.1)\\
\mathrm{degree}(\mathbb{P}; W) &= \mathrm{links}(\mathbb{P}, \mathbb{V}; W) && (4.2)\\
\mathrm{linkratio}(\mathbb{P}, \mathbb{Q}; W) &= \frac{\mathrm{links}(\mathbb{P}, \mathbb{Q}; W)}{\mathrm{degree}(\mathbb{P}; W)}. && (4.3)
\end{aligned}$$

A K-way node partitioning is denoted by $\Gamma_{\mathbb{V}}^{K} = \{\mathbb{V}_1, \cdots, \mathbb{V}_K\}$, where $\mathbb{V}$ is decomposed into K disjoint sets, i.e., $\mathbb{V} = \cup_{l=1}^{K} \mathbb{V}_l$ and $\mathbb{V}_k \cap \mathbb{V}_l = \varnothing$, $k \neq l$. The *normalized associations* and *normalized cuts* criteria are the average of K $\mathrm{linkratio}$'s:

$$\begin{aligned}
\mathrm{knassoc}(\Gamma_{\mathbb{V}}^{K}; W) &= \frac{1}{K} \sum_{l=1}^{K} \mathrm{linkratio}(\mathbb{V}_l, \mathbb{V}_l; W) && (4.4)\\
\mathrm{kncuts}(\Gamma_{\mathbb{V}}^{K}; W) &= \frac{1}{K} \sum_{l=1}^{K} \mathrm{linkratio}(\mathbb{V}_l, \mathbb{V} \setminus \mathbb{V}_l; W). && (4.5)
\end{aligned}$$

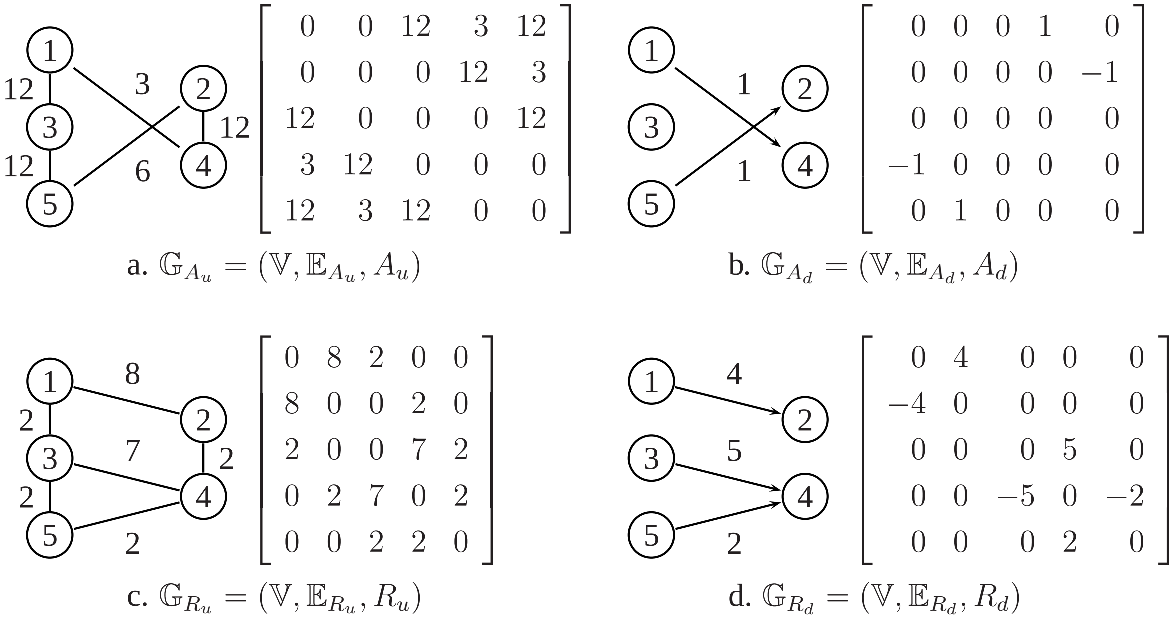
When $W = A$, the associations measure the average within-group similarity and the cuts measure the average between-group similarity.

How do we formulate a partitioning that favors between-group relationships in one direction? There are two issues in defining such a criterion. First, we need to evaluate within-group connections regardless of the asymmetry of internal connections. Secondly, we need to reflect our directional bias on between-group connections.

For these two purposes, we decompose $2\,W$ into a non-directional component W_u and a purely directional component W_d (Fig 4.2):

$$2\,W = W_u + W_d, \quad W_u = (W + W^T), \quad W_d = (W - W^T). \tag{4.6}$$

For each edge, W_u has the sum of the weights in both directions. The total connections for W are thus exactly those of W_u. W_d is a net difference of W between weights attached to edges pointing in opposite directions.



$$\begin{bmatrix} 0 & 0 & 12 & 3 & 12 \\ 0 & 0 & 0 & 12 & 3 \\ 12 & 0 & 0 & 0 & 12 \\ 3 & 12 & 0 & 0 & 0 \\ 12 & 3 & 12 & 0 & 0 \end{bmatrix}$$

a. $\mathbb{G}_{A_u} = (\mathbb{V}, \mathbb{E}_{A_u}, A_u)$

$$\begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

b. $\mathbb{G}_{A_d} = (\mathbb{V}, \mathbb{E}_{A_d}, A_d)$

$$\begin{bmatrix} 0 & 8 & 2 & 0 & 0 \\ 8 & 0 & 0 & 2 & 0 \\ 2 & 0 & 0 & 7 & 2 \\ 0 & 2 & 7 & 0 & 2 \\ 0 & 0 & 2 & 2 & 0 \end{bmatrix}$$

c. $\mathbb{G}_{R_u} = (\mathbb{V}, \mathbb{E}_{R_u}, R_u)$

$$\begin{bmatrix} 0 & 4 & 0 & 0 & 0 \\ -4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & -5 & 0 & -2 \\ 0 & 0 & 0 & 2 & 0 \end{bmatrix}$$

d. $\mathbb{G}_{R_d} = (\mathbb{V}, \mathbb{E}_{R_d}, R_d)$

Figure 4.2: Decomposition of Fig 4.1 into directed and undirected parts.

In Fig 4.2, we can clearly identify $\{1, 3, 5\}$ as the figure and $\{2, 4\}$ as the ground because: there are large within-group connections for A_u, large between-group connections for R_u, and all directed edges pointing from figure to ground.

With directed relationships, we seek an ordered bipartitioning $(\mathbb{V}_1, \mathbb{V}_2)$ that not only has tight connections within groups and loose connections between groups, but most directed edges also go from $\mathbb{V}_1$ to $\mathbb{V}_2$. Below is one choice that meets these conditions:

$$\begin{aligned}\operatorname{knassoc}(A,R;\beta) = \frac{\beta}{2}\cdot\sum_{l=1}^{2}\frac{\operatorname{links}(\mathbb{V}_l,\mathbb{V}_l;A_u)+\operatorname{links}(\mathbb{V}_l,\mathbb{V}\setminus\mathbb{V}_l;R_u)}{\operatorname{degree}(\mathbb{V}_l;A_u+R_u)}+ \\ \frac{1-\beta}{2}\cdot\frac{\operatorname{links}(\mathbb{V}_1,\mathbb{V}_2;A_d+R_d)-\operatorname{links}(\mathbb{V}_2,\mathbb{V}_1;A_d+R_d)}{\sqrt{\operatorname{degree}(\mathbb{V}_1;A_u+R_u)\cdot\operatorname{degree}(\mathbb{V}_2;A_u+R_u)}},\end{aligned} \tag{4.7}$$

$$\begin{aligned}\operatorname{kncuts}(A,R;\beta) = \frac{\beta}{2}\cdot\sum_{l=1}^{2}\frac{\operatorname{links}(\mathbb{V}_l,\mathbb{V}\setminus\mathbb{V}_l;A_u)+\operatorname{links}(\mathbb{V}_l,\mathbb{V}_l;R_u)}{\operatorname{degree}(\mathbb{V}_l;A_u+R_u)}+ \\ \frac{1-\beta}{2}\cdot\frac{\operatorname{links}(\mathbb{V}_2,\mathbb{V}_1;A_d+R_d)-\operatorname{links}(\mathbb{V}_1,\mathbb{V}_2;A_d+R_d)}{\sqrt{\operatorname{degree}(\mathbb{V}_1;A_u+R_u)\cdot\operatorname{degree}(\mathbb{V}_2;A_u+R_u)}}.\end{aligned} \tag{4.8}$$

An interpretation of these definitions is as follows. For the non-directional component, $\operatorname{links}(\mathbb{V}_l,\mathbb{V}_l;A_u)$ are the association by attraction; $\operatorname{links}(\mathbb{V}_l,\mathbb{V}\setminus\mathbb{V}_l;R_u)$ are the segregation by repulsion. We desire both of them to be maximized, after a proper normalization using the degree of the node set.

For directional component, only the difference in connections matters. That is what $\operatorname{links}(\mathbb{V}_1,\mathbb{V}_2;A_d+R_d)-\operatorname{links}(\mathbb{V}_2,\mathbb{V}_1;A_d+R_d)$ measure. Since these cross connections involve two node sets, we normalize them by the geometrical average of their total connections. Similar to $\operatorname{linkratio}(\mathbb{P},\mathbb{Q};W)$, this again is a unit-less connection ratio.

As functions of (A,R), these partitioning criteria favor both attractive and repulsive edges from $\mathbb{V}_1$ to $\mathbb{V}_2$. The directions can also be different for A and R. For example, the ordered partitioning based on $\operatorname{knassoc}(A^T,R;\beta)$ favors repulsion from $\mathbb{V}_1$ to $\mathbb{V}_2$, but attraction from $\mathbb{V}_2$ to $\mathbb{V}_1$.

Finally, β is a parameter modulating the relative importance between undirected and directed graph partitioning. When $\beta=1$, the partitioning ignores the asymmetry in connection weights, while when $\beta=0$, the partitioning only cares about the asymmetry in graph weights. Note that the duality between $\operatorname{knassoc}$ and kncuts is maintained as $\operatorname{knassoc}+\operatorname{kncuts}=\beta$. We will not differentiate the two criteria further and denote our objective as $\varepsilon=\operatorname{knassoc}$.

4.2 Solving an Ordered Partitioning

Let $X = [X_1, X_2]$ be a binary partition matrix, where $X_l(i) = \mathrm{istrue}(i \in \mathbb{V}_l)$. $\mathrm{istrue}(\cdot)$ is a boolean function, which returns 1 if the argument is true and 0 otherwise. Since a node is only assigned to one partition, there is an exclusion constraint on X: $X\,1_2 = 1_N$, where 1_d denotes the $d \times 1$ column vector of 1's.

We define an equivalent degree matrix $\hat{D}$, equivalent weight matrices U and V for the undirected and directed components respectively:

$$\begin{aligned} \hat{D} &= D_{A_u} + D_{R_u} && (4.9) \\ U &= \beta \cdot (A_u - R_u + D_{R_u}) && (4.10) \\ V &= (1-\beta) \cdot (A_d + R_d), && (4.11) \end{aligned}$$

where D_W denotes the degree matrix of W:

$$D_W = \mathrm{Diag}(W 1_N). \tag{4.12}$$

Diag and diag are the conjugate operators that form a diagonal matrix and extract the diagonal of a matrix respectively. Note that U is symmetric: $U = U^T$, and V is skew-symmetric: $V = -V^T$.

With these symbols, we translate Eqn (4.7) into the following program *PNC*:

$$\text{maximize} \quad \varepsilon(X) = \frac{1}{2}\left[\sum_{l=1}^{2} \frac{X_l^T U X_l}{X_l^T \hat{D} X_l} + \frac{X_1^T V X_2 - X_2^T V X_1}{\sqrt{X_1^T \hat{D} X_1 \cdot X_2^T \hat{D} X_2}}\right] \tag{4.13}$$

$$\text{subject to} \quad X \in \{0,1\}^{N \times 2}, \quad X\,1_2 = 1_N. \tag{4.14}$$

In Eqn (4.13), the first term measures the symmetric connections within groups, which is used for an undirected graph partitioning, while the second term favors the connections from $\mathbb{V}_1$ to $\mathbb{V}_2$, which is used for a directed graph partitioning. These two components are taken into account at the same time.

Following a procedure similar to (Yu and Shi, 2003), we solve *PNC* in two steps: first find all global optima in the continuous domain and then find a discrete solution closest to the continuous optima.

4.2.1 Finding Continuous Optima

We introduce a scaled partition matrix Z, which has a one-to-one mapping to X:

$$Z = f(X) = X(X^T \hat{D} X)^{-\frac{1}{2}} \tag{4.15}$$

$$X = f^{-1}(Z) = \mathrm{Diag}(\mathrm{diag}(ZZ^T))^{-\frac{1}{2}} Z. \tag{4.16}$$

We simplify Eqn (4.13) as a function of Z:

$$\varepsilon(Z) = \frac{1}{2} \operatorname{tr} \left(Z^T U Z \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + Z^T V Z \cdot \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \right), \tag{4.17}$$

where tr denotes the trace of a matrix. The 2×2 matrix constants are known to be the matrix representation of the units of complex numbers: 1 and $i = \sqrt{-1}$. Let ^{H} denote the conjugate transpose operator. Noting that $Z^T U Z$ is diagonal and $Z^T V Z$ has zero diagonal, we reduce $\varepsilon(Z)$ further as:

$$\varepsilon(Z) = \frac{1}{2} \operatorname{tr} \left(Z^T U Z \cdot \begin{bmatrix} 1 \\ i \end{bmatrix} \begin{bmatrix} 1 \\ i \end{bmatrix}^H - Z^T V Z \cdot i \cdot \begin{bmatrix} 1 \\ i \end{bmatrix} \begin{bmatrix} 1 \\ i \end{bmatrix}^H \right) \tag{4.18}$$

$$= \frac{1}{2} \left(\begin{bmatrix} 1 \\ i \end{bmatrix}^H Z^T \, U \, Z \begin{bmatrix} 1 \\ i \end{bmatrix} - \begin{bmatrix} 1 \\ i \end{bmatrix}^H Z^T \, iV \, Z \begin{bmatrix} 1 \\ i \end{bmatrix} \right) \tag{4.19}$$

$$= \frac{z^H \hat{W} z}{2}, \tag{4.20}$$

where $\hat{W}$ is the equivalent complex weight matrix:

$$\hat{W} = U - i \cdot V. \tag{4.21}$$

and z is a column vector with a one-to-one mapping to Z:

$$z = g(Z) = Z \begin{bmatrix} 1 \\ i \end{bmatrix} = Z_1 + i \cdot Z_2 \tag{4.22}$$

$$Z = g^{-1}(z) = [\mathrm{re}(z), \mathrm{im}(z)]. \tag{4.23}$$

re and im denote the real and imaginary parts of complex numbers respectively.

In Eqn (4.21), we see that ordering cues complement reciprocal cues along an orthogonal dimension. This generalizes graph partitioning from a symmetric weight matrix to an arbitrary Hermitian weight matrix.

Based on its definition, Z has a natural constraint: $Z^T \hat{D} Z = I$, where I is an identity matrix. This constraint is transferred to z as:

$$z^H \hat{D} z = \begin{bmatrix} 1 \\ i \end{bmatrix}^H Z^T \hat{D} Z \begin{bmatrix} 1 \\ i \end{bmatrix} = \begin{bmatrix} 1 \\ i \end{bmatrix}^H \begin{bmatrix} 1 \\ i \end{bmatrix} = 2. \qquad (4.24)$$

Putting it all together, we have an optimization program in z:

$$\text{maximize} \quad \varepsilon(z) = \frac{1}{2} \cdot z^H \hat{W} z \qquad (4.25)$$

$$\text{subject to} \quad z^H \hat{D} z = 2. \qquad (4.26)$$

Since both $\hat{W}$ and $\hat{D}$ are Hermitian, the Rayleigh-Ritz theorem states that the optimal value of this program is given by the largest generalized eigenvalue of $(\hat{W}, \hat{D})$, and it is achieved by the corresponding eigenvector:

$$\varepsilon(z^*) = \lambda, \quad \hat{W} z^* = \lambda \hat{D} z^*. \qquad (4.27)$$

Assuming that λ is non-repeating, the set of all global optima is given by:

$$\{z = z^* e^{i\theta} : \theta \in [-\pi, \pi]\}, \qquad (4.28)$$

since an eigenvector multiplied by a complex number is still an eigenvector.

Next we transform z back to Z using g^{-1}, and then to X using f^{-1}. Since

$$\begin{aligned} f^{-1}(g^{-1}(z^* e^{i\theta})) &= f^{-1}\left(g^{-1}(z^*) \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \right) \\ &= f^{-1}\left(g^{-1}(z^*)\right) \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}, \end{aligned}$$

the continuous optima in Eqn (4.28) are mapped to:

$$\left\{ \tilde{X}^* R : \tilde{X}^* = f^{-1}(g^{-1}(z^*)),\ R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}, \theta \in [-\pi, \pi] \right\}. \qquad (4.29)$$

In other words, all continuous optima are linked by arbitrary rotations in the 2D plane. Compared to a non-directional graph partitioning (Yu and Shi, 2003), where R is an arbitrary orthogonal matrix, Eqn (4.29) allows rotations but no reflections, through which the identities of $\mathbb{V}_1$ and $\mathbb{V}_2$ as figure and ground are preserved in their relative phases. Ideally, according to Eqn (4.22), ground nodes have a 90° counterclockwise phase advance relative to figure nodes.

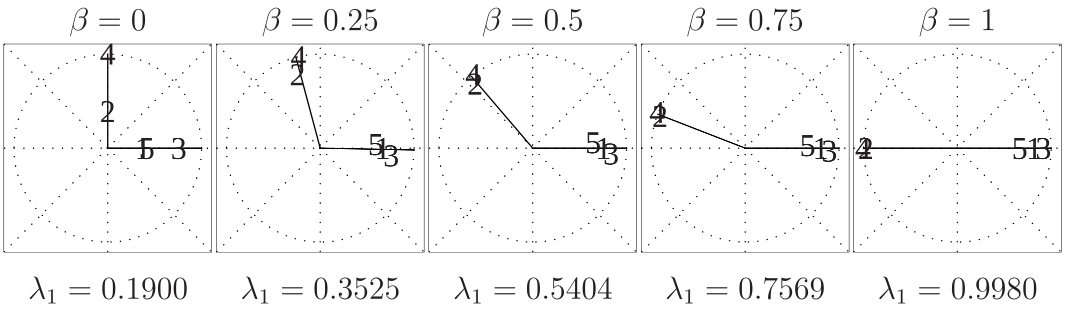


Figure 4.3: Phase encoding of an ordered partitioning. Each plot has 5 points, corresponding to the components of the leading eigenvector of $(\hat{W}, \hat{D})$, where β is varied from 0 to 1. All five cases have two clusters: $\{1, 3, 5\}$ and $\{2, 4\}$, with the latter advancing in phase. The amount of phase difference varies with the amount of directed weights: from 90° for directed weights only, to 180° for undirected edges only.

For Fig 4.2, the equivalent degree matrix and weight matrix with $\beta = 0.5$ are:

$$\hat{D} = \begin{bmatrix} 37 & 0 & 0 & 0 & 0 \\ 0 & 25 & 0 & 0 & 0 \\ 0 & 0 & 35 & 0 & 0 \\ 0 & 0 & 0 & 26 & 0 \\ 0 & 0 & 0 & 0 & 31 \end{bmatrix}, \tag{4.30}$$

$$\hat{W} = \frac{1}{2} \begin{bmatrix} 10 & -8 & 10 & 3 & 12 \\ -8 & 10 & 0 & 10 & 3 \\ 10 & 0 & 11 & -7 & 10 \\ 3 & 10 & -7 & 11 & -2 \\ 12 & 3 & 10 & -2 & 4 \end{bmatrix} + \frac{i}{2} \cdot \begin{bmatrix} 0 & -4 & 0 & -1 & 0 \\ 4 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -5 & 0 \\ 1 & 0 & 5 & 0 & 2 \\ 0 & -1 & 0 & -2 & 0 \end{bmatrix}. \tag{4.31}$$

We expect that the first eigenvector of $(\hat{W}, \hat{D})$ on $\{1, 3, 5\}$ exhibits a phase lag with respect to $\{2, 4\}$. This is verified in Fig 4.3, where the increase of β from 0 to 1 gradually increases the phase lag from $90°$ to $180°$. When the phase lag is $180°$, we can no longer tell which group is the figure.

4.2.2 Finding a Discrete Solution

The second step in solving *PNC* is to get a discrete solution by locating one in the feasible set of *PNC* that is closest to the continuous optima.

Theorem 3 (Optimal Discretization for Ordered Partitioning). *Given $\tilde{X}^* = f^{-1}(g^{-1}(z^*))$, an optimal discrete partition X^* is considered the one satisfying the following program called* POD*:*

$$\begin{aligned} \textit{minimize} \quad & \phi(X, \theta) = \|X - \tilde{X}^* R\|^2 && (4.32) \\ \textit{subject to} \quad & X \in \{0, 1\}^{N \times 2}, \quad X\, 1_2 = 1_N && (4.33) \\ & R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}, \quad \theta \in [-\pi, \pi], && (4.34) \end{aligned}$$

where $\|M\|$ denotes the Frobenius norm of a matrix: $\|M\| = \sqrt{\operatorname{tr}(MM^T)}$. A local optimum (X^*, θ^*) of this bilinear program can be solved iteratively.

Given θ^* hence R^*, POD *is reduced to program* PODX *in X:*

$$\begin{aligned} \textit{minimize} \quad & \phi(X) = \|X - \tilde{X}^* R^*\|^2 && (4.35) \\ \textit{subject to} \quad & X \in \{0, 1\}^{N \times 2}, \quad X\, 1_2 = 1_N. && (4.36) \end{aligned}$$

Let $\tilde{X} = \tilde{X}^* R^*$. The optimal solution is given by non-maximum-suppression:

$$X_1^* = \operatorname{istrue}(\tilde{X}_1 > \tilde{X}_2), \qquad X_2^* = 1 - X_1^*. \qquad (4.37)$$

Given X^*, POD *is reduced to program* PODR *in R hence θ:*

$$\begin{aligned} \textit{minimize} \quad & \phi(\theta) = \|X^* - \tilde{X}^* R\|^2 && (4.38) \\ \textit{subject to} \quad & R = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}, \quad \theta \in [-\pi, \pi], && (4.39) \end{aligned}$$

and the solution can be expressed as the angle of a complex number:

$$\theta^* = \angle\left(g(\tilde{X}^*)^H g(X^*)\right) = \angle\left(\begin{bmatrix}1\\ i\end{bmatrix}^H \tilde{X}^{*T} X^* \begin{bmatrix}1\\ i\end{bmatrix}\right). \tag{4.40}$$

Proof. First we note: $\phi(X,\theta) = \|X\|^2 + \|\tilde{X}^*\|^2 - \operatorname{tr}(XR^T\tilde{X}^{*T} + X^T\tilde{X}^*R) = 2N - 2\operatorname{tr}(XR^T\tilde{X}^{*T})$. Thus minimizing $\phi(X,R)$ is equivalent to maximizing $\operatorname{tr}(XR^T\tilde{X}^{*T})$. For *PODX*, given $R = R^*$, as each entry of $\operatorname{diag}(XR^{*T}\tilde{X}^{*T})$ can be optimized independently, Eqn (4.37) results. For *PODR*, given X^*, we rewrite the objective as

$$\begin{aligned}2\operatorname{tr}(X^*R^T\tilde{X}^{*T}) &= g(X^*)^H g(\tilde{X}^*)e^{i\theta} + g(\tilde{X}^*)^H g(X^*)e^{-i\theta}\\ &= ae^{i\cdot(-\alpha+\theta)} + ae^{i\cdot(\alpha-\theta)}, \quad \text{where} \quad ae^{i\alpha} = g(\tilde{X}^*)^H g(X^*)\\ &= 2a\cos(\theta-\alpha),\end{aligned}$$

which is maximized as $2a$ when $\theta^* = \alpha$. The larger a is, the closer $\tilde{X}^*$ is to X^*. The conclusion results. □

The above discretization procedure has already taken into account the relative phase relationship between groups. The final solution X^* is not just a discrete partitioning, but an ordered partitioning, with X_1^* indicating the figure $\mathbb{V}_1$, and X_2^* indicating the ground $\mathbb{V}_2$.

4.2.3 Algorithm

Given attraction A and repulsion R, given weight β:

1. Decompose graph weights:

$$\begin{aligned}&A_u = A + A^T,\ A_d = A - A^T\\ &R_u = R + R^T,\ R_d = R - R^T.\end{aligned}$$

2. Compute equivalent weight matrix and degree matrix:

$$\hat{W} = \beta \cdot (A_u - R_u + D_{R_u}) - i \cdot (1-\beta) \cdot (A_d + R_d)$$
$$\hat{D} = D_{A_u} + D_{R_u}.$$

3. Find the eigenvector z^* with the largest eigenvalue by: $\hat{W} z^* = \lambda \hat{D} z^*$.
4. Find the corresponding continuous optimal partition matrix:

$$\tilde{X}^* = \mathrm{Diag}(\mathrm{diag}(z^* z^{*H}))^{-\frac{1}{2}} \left[\, \mathrm{re}(z^*), \mathrm{im}(z^*)\right].$$

5. Initialize $\theta^* = \frac{\pi}{4} - \angle \tilde{X}^* \begin{bmatrix} 1 \\ i \end{bmatrix}$ and $\bar{\phi}^* = 0$.
6. Rotate the continuous optimum:

$$\tilde{X} = \tilde{X}^* \begin{bmatrix} \cos(\theta^*) & \sin(\theta^*) \\ -\sin(\theta^*) & \cos(\theta^*) \end{bmatrix}.$$

7. Find the optimal discrete solution X^* by:

$$X_1^* = \mathrm{istrue}(\tilde{X}_1 > \tilde{X}_2),\ X_2^* = 1 - X_1^*.$$

8. Compute the optimal rotation θ^* by:

$$z = \begin{bmatrix} 1 \\ i \end{bmatrix}^H \tilde{X}^{*T} X^* \begin{bmatrix} 1 \\ i \end{bmatrix}$$
$$\bar{\phi} = \|z\|^2$$
If $|\bar{\phi} - \bar{\phi}^*| <$ machine precision, then stop and output X^*
$$\bar{\phi}^* = \bar{\phi}$$
$$\theta^* = \angle z.$$

9. Go to Step 6.

Step 5 rotates the center of $g(\tilde{X}^*)$ to align with 45° so that the next discretization step would split all nodes into two groups. This heuristic is effective but not essential. In Step 6, $\cos(\theta^*)$ and $\sin(\theta^*)$ can also be computed without evaluating θ^* first.

4.3 Experiments

Fig 4.4 and Fig 4.5 show that attraction and repulsion complement each other and their interaction gives a better segmentation. We use spatial proximity for attraction. Since intensity similarity is not considered, we cannot possibly segment this image with attraction alone. Repulsion is determined by relative depths suggested by the T-junction at the center. The repulsion strength falls off exponentially along the direction perpendicular to the T-arms. We can see that attraction alone is not indicative at all since no segmentation cues are encoded. Repulsion only makes boundaries stand out. However, when they work together, repulsion pushes two regions apart at the boundary, while attraction carries this segregation further to the interior of each region thanks to its transitivity.

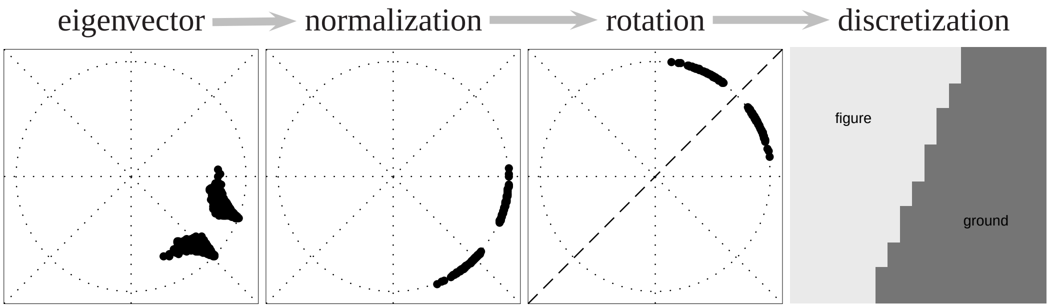


Figure 4.4: Progression of our algorithm. The first plot is the largest eigenvector of $(\hat{W}, \hat{D})$, whose image versions are shown in Row #1-2 of Column #4 in Fig 4.5. Examining the images with the plot, we can identify that the top (bottom) cluster corresponds to the right (left) part of the T-junction. Each point is then normalized to unit lengths. This maps the optimum from the Z- to the X-space. We rotate these points to align their center with the 45° dashed line. A discrete segmentation is obtained by dividing all points with this line: those above it are the ground and those below it are the figure.

Fig 4.6 shows three objects ordered in depth. We compute pairwise affinity based on intensity difference. Partitioning with attraction finds the object of the highest contrast only; with non-directional repulsion, all objects against a common background pop out. If we add in directional repulsion based on occlusion cues, the three objects are further segregated in depth.

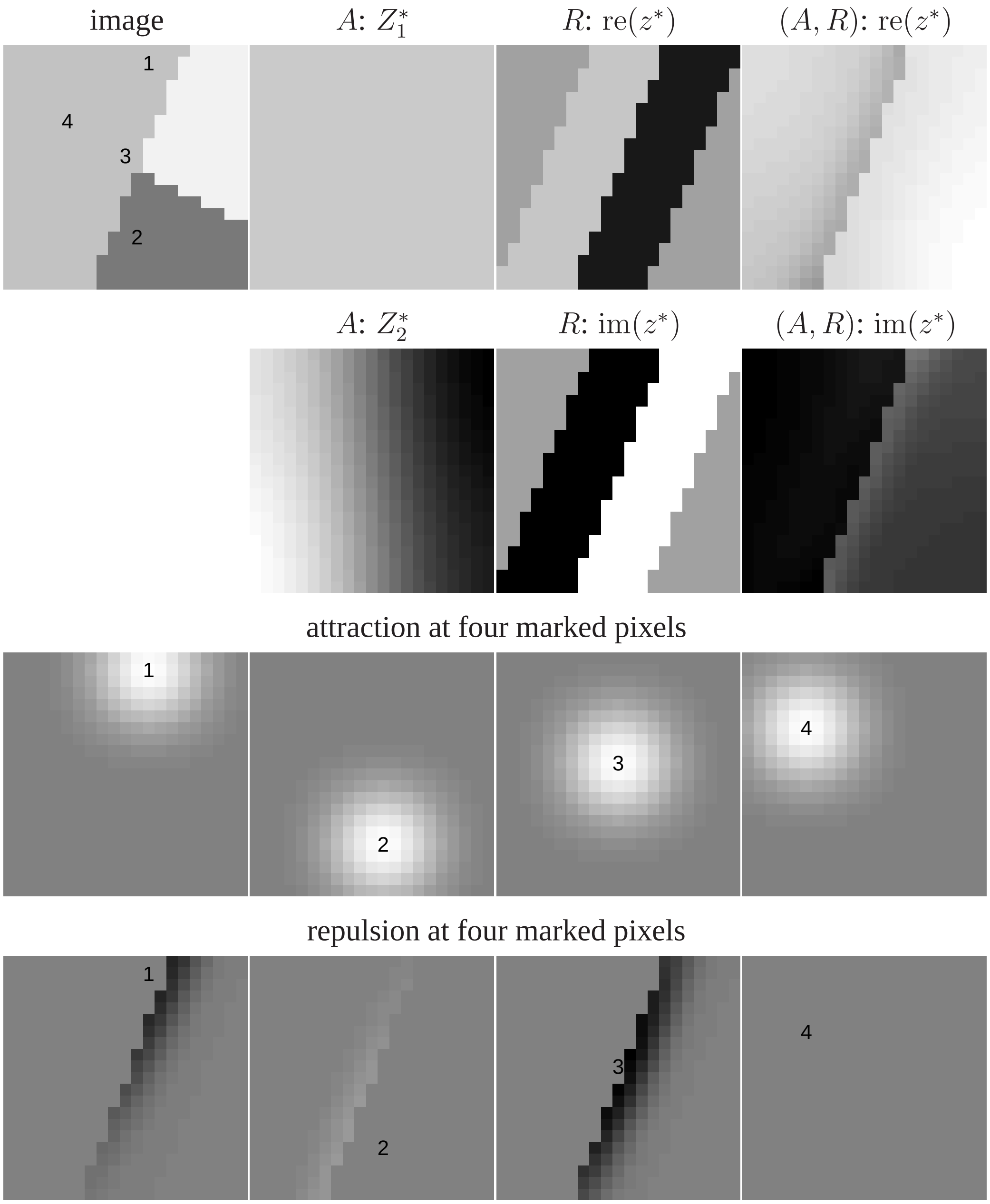


image
A: Z_1^*
R: $\mathrm{re}(z^*)$
(A,R): $\mathrm{re}(z^*)$
1
4
3
2
A: Z_2^*
R: $\mathrm{im}(z^*)$
(A,R): $\mathrm{im}(z^*)$
attraction at four marked pixels
1
2
3
4
repulsion at four marked pixels
1
2
3
4

Figure 4.5: Collaboration of attraction and repulsion. Row #1-2: image and eigenvector results with attraction, repulsion, and both of them. Rows #3-4: attraction and repulsion fields at the four marked locations. Attraction is determined by proximity, thus has the same pattern for all pixels. Repulsion is determined by the T-junction at the center. Most repulsion is zero, while pixels of positive (negative) values are in front of (behind) the marked pixel. Lighter gray for larger values.

Unlike attraction, repulsion is not transitive. If object 3 is in front of object 2, which is in front of object 1, object 3 is not necessarily in front of object 1. We infer from Fig 4.6 that object 1 is in front of object 3 instead, since relative depth between object 3 and 1 is not known to our model.

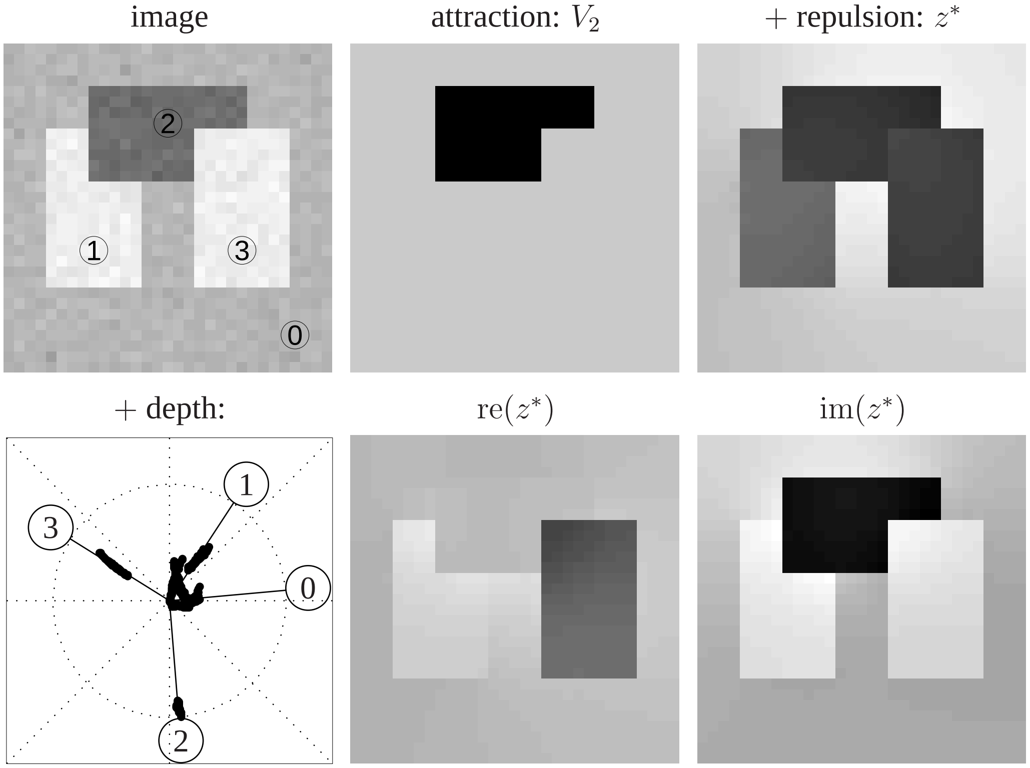


Figure 4.6: Roles of repulsion and depth cues. Image size: 31×31. The background and three objects are marked from 0 to 3, with average intensities of 0.6, 0.9, 0.2 and 0.9. Gaussian noise with standard deviation 0.03 is added. Object 2 has the highest contrast. Attraction and undirected repulsion are computed by the difference of Gaussian ($\sigma_1 = 0.1$, $\sigma_2 = 0.3$) on intensity difference. The neighborhood radius is 2. Row #1: image and results with attraction and undirected repulsion. Row #2: results when directed repulsion from relative depth at the T-junctions is incorporated.

In (Yu and Shi, 2001c), a simple real image example was given where depth ordering was manually derived from T-junctions. However, not only detecting corners and junctions is not well solved in computer vision, but most importantly, T-junctions in real images rarely result from occlusion, hence they do not indicate depth ordering. A good application of our method is yet to be seen.

4.4 Summary

We developed an ordered partitioning method in spectral graph theory. Two directed graphs are used to represent general asymmetric relationships between grouping elements. The weights of the graphs encode either attraction by feature similarity or repulsion by feature dissimilarity. For an ordered bi-partitioning, we desire more connections from one group to the other.

We generalized normalized cuts to such a pair of directed graphs. Our formulation leads to a Rayleigh quotient of a Hermitian matrix, where the imaginary part is the difference of weights on two edges of opposite directions, and the real part is the total weights along both directions, with positive numbers for attraction and negative numbers for repulsion. The global optimum in the continuous domain can be computed by eigendecomposition. An ordered partitioning is then encoded in the phases of the complex eigenvector: the angle separation determines the partitioning, while the relative phase advance indicates the ordering.

We illustrated our method on synthetic image segmentation, where asymmetric repulsion naturally represents relative depth orders arising from occlusion cues. Therefore, surface cues and depth cues can be treated equally in one framework, allowing image segmentation and figure-ground segregation to be computed in one step.

Although our formulation is given for bipartitioning, our formulation in Eqn (4.13) can be extended to $2K$-way directional partitioning:

$$\begin{aligned}
\text{maximize} \quad & \varepsilon(X) = \frac{1}{2K}\left[\sum_{l=1}^{2K} \frac{X_l^T U X_l}{X_l^T \hat{D} X_l} + \sum_{l=1}^{K} \frac{X_l^T V X_{K+l} - X_{K+l}^T V X_l}{\sqrt{X_l^T \hat{D} X_l \cdot X_{K+l}^T \hat{D} X_{K+l}}}\right] \\
\text{subject to} \quad & X \in \{0,1\}^{N\times 2K}, \quad X\,1_{2K} = 1_N.
\end{aligned}$$

That is, we divide all nodes into $2K$ disjoint sets and they form two camps: $\{\mathbb{V}_1, \dots, \mathbb{V}_K\}$ and $\{\mathbb{V}_{K+1}, \dots, \mathbb{V}_{K+K}\}$. The $2K$ sets have tight within-group connections and loose between-group connections in terms of undirected weights, but more connections from the first camp to the other in terms of directed weights, in corresponding nodeset pairs of $(\mathbb{V}_l, \mathbb{V}_{K+l})$, $l = 1, \dots, K$. The continuous optimal solution is then generated by the first K eigenvectors of $(\hat{W}, \hat{D})$, and the discretization procedure can be extended accordingly. We leave the details to the future when such computational models can be employed to solve some real application problems.

Chapter 5

Grouping with Bias

A good image segmentation respects not only the structural properties of the image (Witkin and Tenenbaum, 1983) but also the needs of later visual processing such as object recognition (Xu et al., 2000). In this chapter, we will develop a method that integrates both data-driven and task-driven knowledge for making a global decision on segmentation.

We consider the type of task-driven knowledge presented as *partial grouping* information. For example, in Fig. 5.1, based on intensity distribution and viewers' expectation, we expect a set of bright pixels to be foreground and a set of dark pixels to be background for the image with the tiger, and pixels near image boundaries to be background for the image with the fashion-model. Such information provides *bias* to a natural grouping process based solely on data themselves.

In our work, we are concerned with the following issue: what is a simple and principled approach for incorporating these often *sparse* partial grouping cues directly into low-level image segmentation?

A straightforward answer to this question that we adopt in our work is to formulate it as a constrained optimization problem, where the goodness of a segmentation is based on low-level data coherence, and the feasibility of a segmentation is based on partial grouping constraints. For the normalized cuts criterion (Shi and Malik, 1997) in spectral graph theory, we show that this straightforward formulation leads to a constrained eigenvalue problem. By generalizing the standard

Rayleigh-Ritz theorem, we can compute a near-global optimum efficiently.

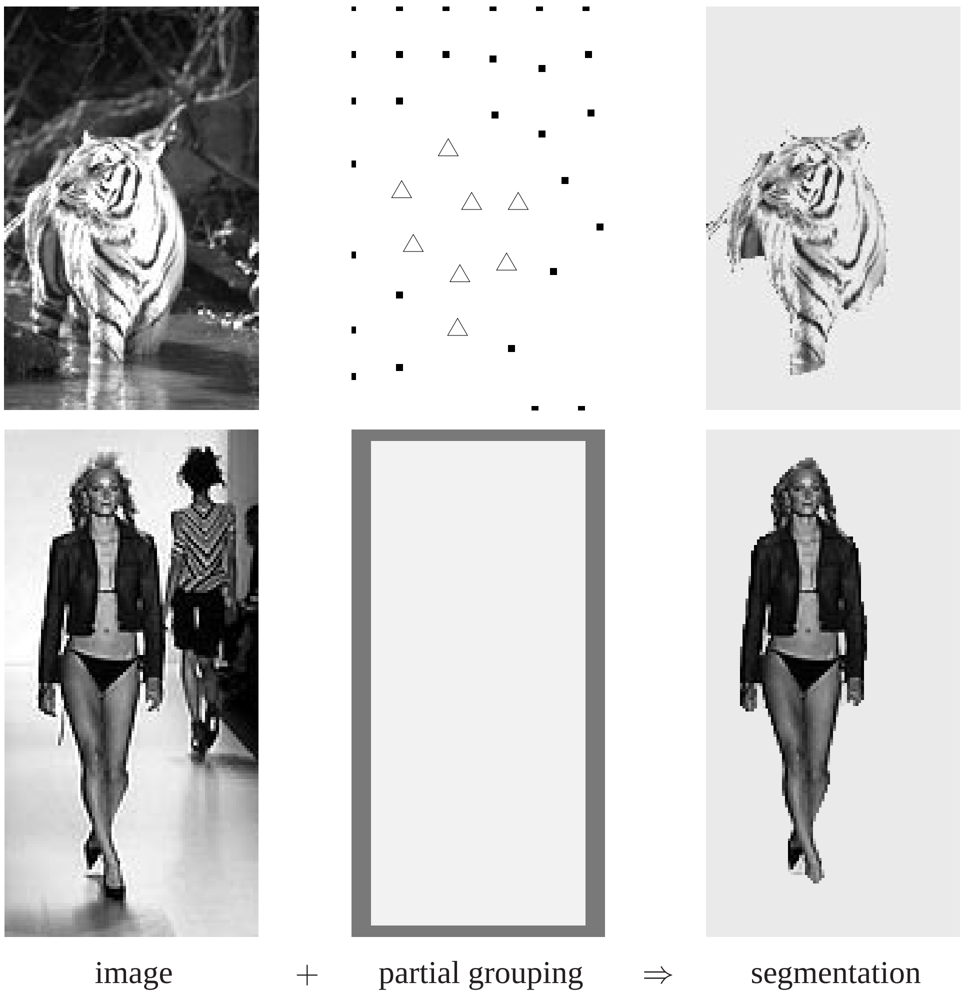


Figure 5.1: Segmentation under partial grouping constraints. We desire an algorithm that integrates partial grouping cues with natural grouping based on data coherence and outputs an object segmentation. In the middle column, white pixels are unlabeled, whereas marked or gray pixels are *a priori* known to be in the same group. These cues are derived from feature-driven or location-driven attentional maps. That is, the regions of interest here are defined based on pixel intensities or prior expectation of object locations.

We then show through a simple point set example that segmentation performance breaks down especially when partial grouping cues are sparse. This observation leads to a new formulation with smoothed constraints. In the spectral graph framework, the smoothing operator is readily derived from the existing pairwise relationships between grouping elements. We present numerous image segmentation examples to demonstrate the efficacy of the new formulation. Finally, we conclude with a discussion on the connections to related data clustering methods.

5.1 Basic Formulation

Given an image of N pixels, the goal of segmentation is to assign one of K prescribed labels to each pixel. Let $\mathbb{V} = [N]$ denote the set of all pixels, where $[n]$ denotes the set of integers between 1 and n: $[n] = \{1, 2, \ldots, n\}$. To segment an image is to decompose $\mathbb{V}$ into K disjoint sets, i.e., $\mathbb{V} = \cup_{l=1}^{K} \mathbb{V}_l$ and $\mathbb{V}_k \cap \mathbb{V}_l = \varnothing$, $k \neq l$. We denote this K-way partitioning by $\Gamma_{\mathbb{V}}^{K} = \{\mathbb{V}_1, \ldots, \mathbb{V}_K\}$.

Let $\varepsilon(\Gamma_{\mathbb{V}}^{K}; f)$ be an objective function that measures the goodness of grouping for the image data f, e.g. $f(i)$ is the intensity value at pixel i, $i \in \mathbb{V}$. In Markov random field (MRF) approaches for image segmentation (Geman and Geman, 1984), the objective function is the posterior probability of the segmentation $\Gamma_{\mathbb{V}}^{K}$ given the observation f:

$$\varepsilon_{MRF}(\Gamma_{\mathbb{V}}^{K}; f) = \Pr(\Gamma_{\mathbb{V}}^{K} | f) = \Pr(f | \Gamma_{\mathbb{V}}^{K}) \cdot \Pr(\Gamma_{\mathbb{V}}^{K}). \tag{5.1}$$

The first term $\Pr(f | \Gamma_{\mathbb{V}}^{K})$ describes data fidelity, which measures how well a generative model explains the observed image data, and the second term $\Pr(\Gamma_{\mathbb{V}}^{K})$ describes model complexity, which favors the segmentation to have some regularity such as piecewise constancy. In discriminative approaches for segmentation (Shi and Malik, 2000), the objective function is some clustering measure which increases with within-group feature similarity and decreases with between-group feature similarity.

Consider partial grouping information represented by n pixel sets: $\mathbb{U}_t$, $t \in [n]$, each containing pixels known to belong together. However, we do not know which labels should be assigned to these sets of pixels, nor do we require these groups to take distinct labels. For a unique representation of $\mathbb{U}_t$'s, we assume there is no common pixel between any two sets: $\mathbb{U}_s \cap \mathbb{U}_t = \varnothing$, $s \neq t$. In other words, if there is a common pixel, then the two sets should be merged into one.

The most straightforward way to incorporate the partial grouping information is to encode it as constraints. With a little abuse of notation, we use $\Gamma_{\mathbb{V}}^{k}(i,l)$ to denote a boolean function that returns 1 if $i \in \mathbb{V}_l$. Among the segmentations partially determined by $\mathbb{U}_t$'s, we seek one that optimizes the goodness of grouping measured by ε:

$$\text{maximize } \varepsilon(\Gamma_{\mathbb{V}}^{K}; f) \tag{5.2}$$

$$\text{subject to } \Gamma_{\mathbb{V}}^{K}(i,l) = \Gamma_{\mathbb{V}}^{K}(j,l),\ i,j \in \mathbb{U}_t, l \in [K],\ t \in [n]. \tag{5.3}$$

Since partial grouping cues are encoded as hard constraints, they have to be reliable enough to be enforced. Fig. 5.1 illustrates two basic scenarios where we can derive such cues. The first type is feature-driven, where pixels conforming to a particular generative model are biased together. For example, we probably perceive a white object against a dark background before we realize that it is a tiger in a river. In this case, $\mathbb{U}_1$ contains pixels of the brightest intensities and $\mathbb{U}_2$ the darkest. The second type is solely location-driven, it reflects our expectation as to where an object is going to appear. For example, pictures taken in a fashion show often have fashion models at the center. To segment out the fashion models, we consider pixels at image boundaries in $\mathbb{U}_1$ as the background group. Such seemingly insignificant information provides long-range binding cues which are often lacking in low-level grouping.

For some particular forms of ε, such as the above mentioned probability criteria using generative models and minimum cuts criteria in discriminative approaches (Ishikawa and Geiger, 1998; Roy and Cox, 1998; Boykov et al., 1999), the constraints in Eqn (5.3) can be trivially incorporated in an algorithm that optimizes the objective. For the former, Markov Chain Monte Carlo is a general solution technique and the constraints can be realized by generating legitimate samples (Zhu, 1999). For the latter, assuming that $\mathbb{U}_1$ and $\mathbb{U}_2$ take distinct labels, we can solve Eqn (5.3) using maximum-flow algorithms, in which two special nodes called source and sink are introduced, with infinite weights set up between the source and nodes in $\mathbb{U}_1$, the sink and nodes in $\mathbb{U}_2$ (Ishikawa and Geiger, 1998). For others such as the normalized cuts criterion (Shi and Malik, 2000), it is not clear whether the solution can be obtained using the same technique as the unconstrained problem. We will explore this criterion further.

5.2 Constrained Normalized Cuts Criterion

A weighted graph is specified by $\mathbb{G} = (\mathbb{V}, \mathbb{E}, W)$, where $\mathbb{V}$ is the set of all nodes, $\mathbb{E}$ is the set of edges connecting nodes, and W is an affinity matrix, with weights characterizing the likelihood that two nodes belong to the same group.

In graph theoretic methods for image segmentation, an image is first transcribed into a weighted graph, where each node represents a pixel, and weights on edges connecting two nodes describe the pairwise feature similarity between the pixels. Segmentation then becomes a node partitioning problem. A good segmentation desires a partitioning that has tight connections within partitions and loose connections across partitions. These two goals can both be achieved in the normalized cuts criterion (Shi and Malik, 2000). We give a brief self-contained account of this criterion below.

5.2.1 Representation

Given weight matrix W, the multiclass normalized cuts criterion tries to maximize the average of all K $\mathrm{linkratio}$'s (Yu and Shi, 2003):

$$\varepsilon_{NC}(\Gamma_{\mathbb{V}}^{K}) = \frac{1}{K}\sum_{l=1}^{K}\mathrm{linkratio}(\mathbb{V}_l) == \frac{1}{K}\sum_{l=1}^{K}\frac{\sum_{i\in\mathbb{V}_l, j\in\mathbb{V}_l} W(i,j)}{\sum_{i\in\mathbb{V}_l, j\in\mathbb{V}} W(i,j)}. \tag{5.4}$$

The $\mathrm{linkratio}$ is the fraction of the total weights within a group to the total weights all the member nodes have. This criterion favors both tight connections within partitions and loose connections between partitions.

We use an $N \times K$ *partition matrix* X to represent $\Gamma_{\mathbb{V}}^{K}$, where $X = [X_1, \ldots, X_K]$ and $X(i,l) = 1$ if $i \in \mathbb{V}_l$ and 0 otherwise. X_l is a binary indicator for partition

$\mathbb{V}_l$. Since a node is only assigned to one partition, there is an exclusion constraint on X: $X\,1_K = 1_N$, where 1_d denotes the $d \times 1$ vector of all 1's.

For $t \in [n]$, partial grouping node set $\mathbb{U}_t$ produces $|\mathbb{U}_t| - 1$ independent constraints, where $|\cdot|$ denotes the size of a set. Each constraint can be represented by an $N \times 1$ vector U_k with only two non-zero elements: $U_k(i) = 1$, $U_k(j) = -1$, $i, j \in \mathbb{U}_t$ for instance. Let $U = [U_1, \ldots, U_{\bar{n}}]$, where $\bar{n} = \sum_{t=1}^{n}(|\mathbb{U}_t| - 1)$. Then the partial grouping constraints in Eqn (5.3) become: $U^T X = 0$. U obtained as such has a full rank.

Finally, we introduce the degree matrix D, defined to be the total connections each node has: $D = \mathrm{Diag}(W 1_N)$, where Diag denotes a diagonal matrix formed from its vector argument.

With these symbols and notation, we write the constrained grouping problem in Eqn (5.3) for the normalized cuts criterion as program *PNCX*:

$$\text{maximize} \quad \varepsilon_{NC}(X) = \frac{1}{K} \sum_{l=1}^{K} \frac{X_l^T W X_l}{X_l^T D X_l} \tag{5.5}$$

$$\text{subject to} \quad X \in \{0,1\}^{N \times K}, \quad X\,1_K = 1_N \tag{5.6}$$

$$U^T X = 0. \tag{5.7}$$

5.2.2 Computational Solution

We introduce a *scaled partition matrix* Z to make Eqn (5.5) more manageable:

$$Z = X(X^T D X)^{-\frac{1}{2}}. \tag{5.8}$$

Then $\varepsilon_{NC}(X) = \frac{1}{K}\,\mathrm{tr}(Z^T W Z)$, where tr denotes the trace of a matrix. Given the definition in Eqn (5.8), Z naturally satisfies: $Z^T D Z = I$, where I is an identity matrix. The grouping constraint in Eqn (5.7) is equivalent to:

$$U^T Z = U^T X (X^T D X)^{-\frac{1}{2}} = 0. \tag{5.9}$$

Ignoring Eqn (5.6) for the time being, we relax *PNCX* into program *PNCZ*:

$$\begin{aligned} \text{maximize} \quad & \varepsilon_{NC}(Z) = \frac{1}{K} \operatorname{tr}(Z^T W Z) && (5.10) \\ \text{subject to} \quad & Z^T D Z = I && (5.11) \\ & U^T Z = 0. && (5.12) \end{aligned}$$

PNCZ is a constrained eigenvalue problem (Gander et al., 1989) in the continuous domain and it can be solved by linear algebra.

In principle, we can solve *PNCZ* by applying the standard Rayleigh-Ritz theorem to its unconstrained version:

$$\begin{aligned} \text{maximize} \quad & \varepsilon_{NC}(Y) = \frac{1}{K} \operatorname{tr}(Y^T W^y Y) && (5.13) \\ \text{subject to} \quad & Y^T D^y Y = I, && (5.14) \end{aligned}$$

where Y is an $(N - \bar{n}) \times K$ coefficient matrix of Z using an orthonormal basis $U^\perp$ of the feasible solution space, i.e.,

$$Z = U^\perp Y, \qquad U^T U^\perp = 0, \tag{5.15}$$

and $W^y = (U^\perp)^T W U^\perp$ and $D^y = (U^\perp)^T D U^\perp$ are the equivalent weight and degree matrices for Y. This is a standard Rayleigh quotient optimization problem. If (V^y, S^y) is the eigendecomposition of the matrix pair (W^y, D^y), where $S^y = \operatorname{Diag}(s^y)$ with nonincreasingly ordered eigenvalues, then the global optimum is given by the eigenvectors corresponding to the first K largest eigenvalues, and

$$\varepsilon_{NC}([V_1^y, \ldots, V_K^y]) = \frac{1}{K} \sum_{l=1}^{K} s^y(l) = \max_{Y^T D^y Y = I} \varepsilon_{NC}(Y). \tag{5.16}$$

From Eqn (5.15), we recover the global optimum $Z^* = U^\perp [V_1^y, \ldots, V_K^y]$.

The introduction of Y gets rid of the constraint in Eqn (5.12) and turns program *PNCZ* into an unconstrained eigenvalue problem. However, it requires finding an orthonormal basis for the feasible space first. Given that $\bar{n} \ll N$, this process has a space and time complexity of $O(N^2)$ and $O(N^3)$ respectively, which is prohibitively expensive for a large N. We have to find another way out.

There is such an alternative through the use of matrix projectors. Q is called a *projector* if it is *idempotent*, i.e., $Q^2 = Q$. If Q is a projector onto the space of feasible solutions of *PNCZ*, then QZ is the projection of Z on the feasible space. The key property of QZ is that $QZ = Z$ if and only if Z is feasible. Therefore, we can guarantee the feasibility of a solution by projecting it to the feasible set in the original space without resorting to any re-parameterization in a reduced space.

We introduce a few symbols to simplify notation. Let π be any set of K distinct integers from $[N]$. For any eigenvector matrix V and its corresponding eigenvalue matrix $S = \mathrm{Diag}(s)$, let $V_\pi = [V_{\pi_1}, \ldots, V_{\pi_K}]$ and $S_\pi = \mathrm{Diag}([s_{\pi_1}, \ldots, s_{\pi_K}])$.

Theorem 4 (Generalized Rayleigh-Ritz Theorem). *Let (V, S) be the eigendecomposition of matrix QPQ, where P is the row-normalized weight matrix and Q is a projector onto the feasible solution space:*

$$QPQV = VS, \quad S = \mathrm{Diag}(s) \tag{5.17}$$

$$V^T DV = I \tag{5.18}$$

$$P = D^{-1}W \tag{5.19}$$

$$Q = I - D^{-1}U(U^T D^{-1}U)^{-1}U^T. \tag{5.20}$$

For any local optimum candidate Z^* to program PNCZ, *there exists an index set π and an orthonormal matrix R such that:*

$$Z^* = V_\pi R, \quad R^T R = I \tag{5.21}$$

$$\varepsilon(Z^*) = \frac{1}{K}\,\mathrm{tr}(S_\pi). \tag{5.22}$$

Assuming that the eigenvectors are ordered according to their eigenvalues, where $s_1 \geq \cdots \geq s_N$, any global optimum of PNCZ *can thus be specified by the first K largest eigenvectors and an orthonormal matrix:*

$$Z^* = V_{[K]}, \quad R^T R = I \tag{5.23}$$

$$\varepsilon(Z^*) = \frac{1}{K}\,\mathrm{tr}(S_{[K]}) = \max_{\substack{Z^T DZ = I \\ U^T Z = 0}} \varepsilon(Z). \tag{5.24}$$

Proof. We define a *Lagrangian* for *PNCZ*:

$$L(Z, \Lambda, \Theta) = \frac{1}{2}\operatorname{tr}(Z^T W Z) - \frac{1}{2}\operatorname{tr}(\Lambda^T(Z^T D Z - I)) - \Theta^T U^T Z,$$

where Λ is a $K \times K$ symmetric matrix and Θ is an $\bar{n} \times K$ matrix. An optimal solution $(Z^*, \Lambda^*, \Theta^*)$ must satisfy:

$$L_Z(Z, \Lambda, \Theta) = WZ - DZ\Lambda - U\Theta = 0, \tag{5.25}$$

$$L_\Lambda(Z, \Lambda, \Theta) = Z^T D Z - I = 0, \tag{5.26}$$

$$L_\Theta(Z, \Lambda, \Theta) = U^T Z = 0. \tag{5.27}$$

Multiplying Eqn (5.25) with $U^T D^{-1}$ leads to:

$$\Theta^* = (U^T D^{-1} U)^{-1} U^T D^{-1} W Z^*, \tag{5.28}$$

where D and $U^T D^{-1} U$ are invertible since both D and U assume full ranks. Eliminating Θ in Eqn (5.25) by Eqn (5.28), we obtain

$$QPZ^* = Z^* \Lambda^*. \tag{5.29}$$

From Eqn (5.27), we also have $QZ^* = Z^*$. Substituting it into the above equation, we obtain $QPQZ^* = Z^*\Lambda^*$. Therefore, there are three necessary conditions for the optimality: Λ^* is symmetric and

$$QPQZ^* = Z^*\Lambda^*, \quad Z^{*T} D Z^* = I. \tag{5.30}$$

Next we show that there exists an eigendecomposition (V, S) of QPQ that not only meets these conditions but can also generate all such solutions through orthonormal matrices.

Noting that $QPQZ^* = Z^*\Lambda^*$ is equivalent to:

$$D^{\frac{1}{2}} Q D^{-\frac{1}{2}} \cdot D^{\frac{1}{2}} P D^{-\frac{1}{2}} \cdot D^{\frac{1}{2}} Q D^{-\frac{1}{2}} \cdot D^{\frac{1}{2}} Z^* = D^{\frac{1}{2}} Z^* \cdot \Lambda^*, \tag{5.31}$$

we rewrite Eqn (5.30) using a transformed variable $\bar{Z}$:

$$\bar{Q}\bar{P}\bar{Q}\bar{Z} = \bar{Z}\Lambda^*, \quad \bar{Z}^T \bar{Z} = I, \tag{5.32}$$

$$\bar{Z} = D^{\frac{1}{2}} Z^* \tag{5.33}$$

$$\bar{P} = D^{\frac{1}{2}} P D^{-\frac{1}{2}} = D^{-\frac{1}{2}} W D^{-\frac{1}{2}} \tag{5.34}$$

$$\bar{Q} = D^{\frac{1}{2}} Q D^{-\frac{1}{2}} = I - D^{-\frac{1}{2}} U (U^T D^{-1} U)^{-1} U^T D^{-\frac{1}{2}}. \tag{5.35}$$

Since both $\bar{P}$ and $\bar{Q}$ are symmetric, $\bar{Q}\bar{P}\bar{Q}$ is symmetric, which means that all its eigenvectors are real and orthogonal. Therefore, if $(\bar{V}, S)$ is an orthonormal eigendecomposition of $\bar{Q}\bar{P}\bar{Q}$, then any distinct K eigenvectors and their eigenvalues form a solution, $(\bar{V}_\pi, S_\pi)$, to Eqn (5.32).

If $(\bar{Z}, \Lambda^*)$ is a solution that satisfies Eqn (5.32) with $\bar{Z}$ orthonormal and Λ^* symmetric, since $\bar{V}$ is a complete basis in the N-dimensional space, there exists an index set π and an orthonormal matrix R such that

$$\bar{Z} = \bar{V}_\pi R, \quad R^T R = I \tag{5.36}$$

$$\Lambda^* = R^T S_\pi R. \tag{5.37}$$

Multiplying Eqn (5.25) with Z^{*T} and using $\operatorname{tr}(AB) = \operatorname{tr}(BA)$, we derive:

$$\varepsilon_{NC}(Z^*) = \frac{1}{K}\operatorname{tr}(Z^{*T} W Z^*) = \frac{1}{K}\operatorname{tr}(\Lambda^*) = \frac{1}{K}\operatorname{tr}(S_\pi). \tag{5.38}$$

Therefore, $\{(\bar{V}_\pi, S_\pi) : \pi\}$ produce all possible local optimal values. The global optimal value is thus given by the average of the first K largest eigenvalues. Transforming $\bar{Z}$ back to the Z space based on Eqn (5.33), we have $V = D^{-\frac{1}{2}}\bar{V}$ and S as exactly an eigendecomposition of QPQ. This completes the proof. □

When there is no constraint, $Q = I$, then $QPQ = P$ can be considered as a transition probability matrix of random walks, and the normalized cuts criterion is equivalent to a maximum conductance problem where subsets of states only occasionally visit each other (Meila and Shi, 2001). When there are constraints, $Q \neq I$, QPQ usually has negative entries and it no longer has a transition probability interpretation. In other words, the solution to constrained grouping can no longer be cast as the equilibrium of a natural diffusion process.

To summarize, the optimal solution to ***PNCZ*** is not unique. It is a subspace spanned by the first K largest eigenvectors of QPQ by orthonormal matrices:

$$Z^* \in \{V_{[K]} R : QPQ V_{[K]} = V_{[K]} S_{[K]}, R^T R = I\}. \tag{5.39}$$

Unless all K eigenvalues are the same, $V_{[K]}R$ are no longer the eigenvectors of QPQ. Yet all these solutions have the optimal objective value.

After we compute $(V_{[K]}, S_{[K]})$ from QPQ, the same procedure for the unconstrained normalized cuts (Yu and Shi, 2003) can be applied to find a near global-optimal discrete solution to *PNCX*. The only difference is that now the eigenvectors are from QPQ rather than P.

5.2.3 Algorithm

Given weight matrix W and constraint matrix U, below is the algorithm to find the optimal eigensolution $V_{[K]}$ for constrained K-way normalized cuts. A final segmentation can be obtained using the same discretization procedure for unconstrained solutions (Yu and Shi, 2003).

$$\begin{aligned}
&D = \mathrm{Diag}(W 1_N)\\
&\bar{P} = D^{-\frac{1}{2}} W D^{-\frac{1}{2}}\\
&\bar{U} = D^{-\frac{1}{2}} U\\
&H = (\bar{U}^T \bar{U})^{-1}\\
&(I - \bar{U} H \bar{U}^T) \bar{P} (I - \bar{U} H \bar{U}^T) \bar{V}_{[K]} = \bar{V}_{[K]} S_{[K]}, \quad \bar{V}_{[K]}^T \bar{V}_{[K]} = I\\
&V_{[K]} = D^{-\frac{1}{2}} \bar{V}_{[K]}.
\end{aligned}$$

We avoid directly computing $\bar{Q}\bar{P}\bar{Q}$ since it can become a dense matrix with even sparse U and P. Specifically, we modify the innermost iteration formula in an eigensolver. For that, we only need to precompute $\bar{U} = D^{-\frac{1}{2}} U$, which is as sparse as U, and $H = (\bar{U}^T \bar{U})^{-1}$, which is an $\bar{n} \times \bar{n}$ matrix. $\bar{U}$ and H are the only two other matrices apart from those already used for unconstrained cuts. During each iteration of $x := \bar{Q}\bar{P}\bar{Q}x$, we compute:

$$z := \bar{Q}x = x - \bar{U} H \bar{U}^T x \tag{5.40}$$

$$y := \bar{P} z \tag{5.41}$$

$$x := \bar{Q}y = y - \bar{U} H \bar{U}^T y. \tag{5.42}$$

If $\bar{P}$ has an average of k non-zeros per row, then Eqn (5.41) has $O(Nk)$ multiplications. Eqn (5.40) and Eqn (5.42) each requires $O(2N\bar{n} + \bar{n}^2)$ multiplications, and they are the extra computation needed for constrained cuts. Given that $\bar{n} \ll N$ but comparable to k, the increase in time complexity is linear. However, since the solution space is reduced, fewer iterations are needed to converge to the largest eigenvectors. Therefore, the net increase in the computational space and time is negligible if the number of constraints $\bar{n}$ is small. We can reduce the complexity further by sampling the constraints.

5.3 Propagating Constraints

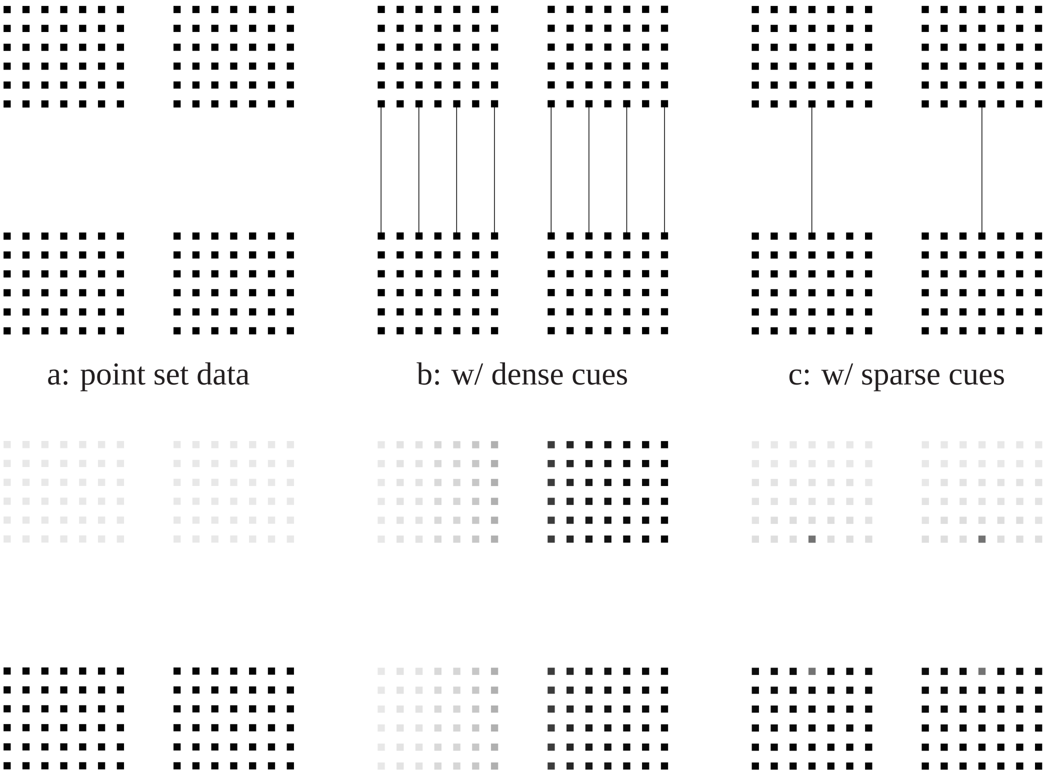
The basic formulation works reasonably well if there are enough partial grouping cues. This is not very useful since in reality only a few such cues are given. Sparse cues expose an inherent flaw in the formulation, however, there is a remedy to it.

5.3.1 Point Set Example

In Fig 5.2, points are naturally organized into four clusters based on proximity. Since the vertical gap is larger than the horizontal gap, an ideal 2-class clustering is obtained by a horizontal cut that divides the four clusters into top and bottom groups. Now if a few points at the horizontal boundary are grouped together *a priori*, the horizontal cut violates the partial grouping constraints and the vertical cut becomes optimal. However, when the number of grouping cues is reduced, the formulation in Eqn (5.3) fails to produce the desired vertical cut that divides the four clusters into left and right groups. In particular, the labeled points tend to stand out, while having little impact on the grouping of the rest points.

5.3.2 Why Simple Constraints Are Insufficient

When we pre-assign points from top and bottom clusters together, we do not just want a group to lose its labeled points to the other group (Fig 5.2c), but rather we desire a grouping process that explores their neighbouring connections and discovers the left-right division instead.



a: point set data

b: w/ dense cues

c: w/ sparse cues

V_2 for a

V_2 for b

V_2 for c

Figure 5.2: Three grouping scenarios illustrating the problem of integrating sparse grouping cues into a grouping engine. Row #1: 12×14 dots with a minimum inter-point distance of 1. Pairs of linked points are known to belong together. The weights are computed using a Gaussian function of distance with a standard deviation of 3. Row #2: the continuous optimum V_2 for normalized cuts. For sparse grouping cues, we no longer have the desired vertical cut as the optimal solution.

The formulation in Eqn (5.3), however, does not entail the desire of propagating grouping information on the constrained data points to their neighbours. Often, a slightly perturbed version of the optimal unbiased segmentation becomes the legitimate optimum (Fig 5.3).

There are two reasons for such a solution to be undesirable. First, the solution is not smooth. One of the biased data points takes a label that is very different

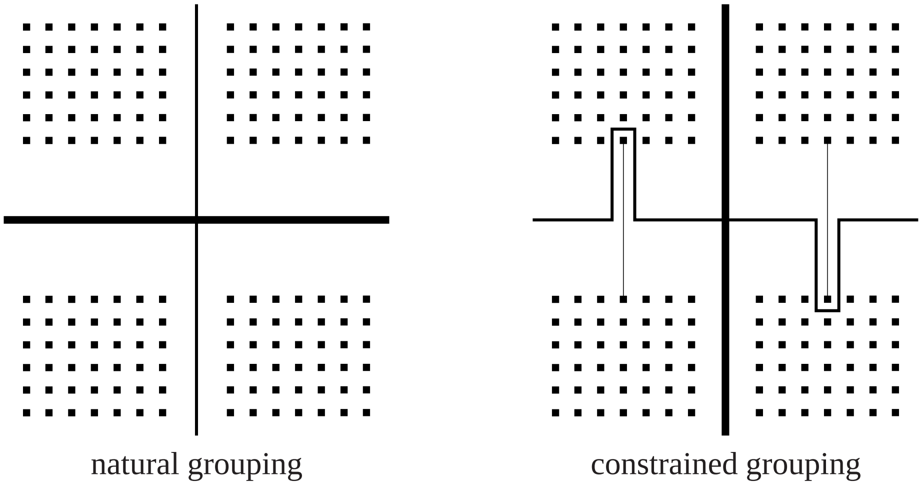


Figure 5.3: Undesired grouping from sparse constraints. In the 2-class grouping based on proximity, the horizontal division is optimal while the vertical division is suboptimal. When two pairs of points from top and bottom groups are linked together, the vertical division is desired. However, perturbation on the unconstrained optimum can lead to a partitioning that satisfies the constraints while producing the maximum objective value ε.

from its nearby points. This is not acceptable especially to those neighbours with which it has high affinity. In other words, we need to explicitly encode *data-driven smoothness* into our discriminative formulation.

The second reason is that such a biased grouping lacks fairness with regard to labeled points. Intuitively, if two labeled points, i and j, have similar connections to their neighbours, we desire a fair segmentation so that if i gets grouped with i's friends, j also gets grouped with j's friends. In Fig 5.3, the two points in a labeled pair have similar affinity patterns to their nearby points, yet their local segmentations are dissimilar in any solution resulting from the perturbation of the unbiased optimal grouping.

These two conditions, smoothness and fairness of a segmentation on a pair of labeled data points, can condition a grouping to the extent that many trivial near-optimal unbiased grouping solutions are ruled out from the feasible solution space. Formally, rather than strictly enforcing equal labels on the biased data points themselves, we desire their average labels to be the same. The average labels take the labels of other data points into account. The more similar a data point is to the biased ones, the heavier the weight is for the label corresponding to the point in the average. Thus the biased data points are prevented from taking a label that is different from what other similar data points have.

Let $g \circ f$ be the compound function of g and f. Let S_f denote a smoothing function contingent on data f. We modify the formulation in Eqn (5.3) to be:

$$\begin{aligned} &\text{maximize} \quad \varepsilon(\Gamma_{\mathbb{V}}^{K}; f) \\ &\text{subject to} \quad S_f \circ \Gamma_{\mathbb{V}}^{k}(i, l) = S_f \circ \Gamma_{\mathbb{V}}^{k}(j, l), i, j \in \mathbb{U}_t,\ l \in \mathbb{K}, t \in [n]. \end{aligned} \tag{5.43}$$

The observation we made in Fig 5.3, the need of propagating constraints, stands on a universal ground from the optimization point of view. Therefore, it holds for all choices of ε. The basic formulation, although straightforward, is inherently flawed.

Our new formulation is not equivalent to the introduction of smoothness priors in a generative approach. There, prior knowledge such as piecewise constancy is usually imposed on the solution independently of goodness of fit (Geman and Geman, 1984), whereas ours is closely coupled with the data coherence. Our essential message in this regard is that an effective propagation of priors requires an intimate interaction with data themselves.

5.3.3 Smooth Constraints for Normalized Cuts

A natural choice of S_f for normalized cuts is the normalized weight matrix P:

$$S_f \circ \Gamma_{\mathbb{V}}^{k}(i, l) = \sum_{j} P_{ij} X(j, l), \quad i \in \mathbb{V},\ l \in [K]. \tag{5.44}$$

This value measures the average density of $\mathbb{V}_l$ from node i's point of view, with nodes of high affinity to it weighted more in the density. This discourages i to take a label different from those of its close neighbours. We may not know in advance what this density is for the optimal partitioning, but the fairness condition requires it to be the same for the labeled pair (i, j): $S_f \circ \Gamma_{\mathbb{V}}^K(i, l) = S_f \circ \Gamma_{\mathbb{V}}^k(j, l)$. The partial grouping constraints in Eqn (5.7) then become:

$$U^T P X = (P^T U)^T X = 0. \tag{5.45}$$

Since the only change here is that the constraint matrix U becomes $P^T U$, the same solution technique applies. That is, the eigensolution to the program *PNCZ* is given by the eigenvectors of QPQ, where Q is a projector onto the solution space specified by $(P^T U)^T X = 0$ instead of $U^T X = 0$.

In Fig 5.4, we show new results with the smoothed constraints. In addition to the basic results in Fig 5.2, we also consider two other alternatives that directly utilize partial grouping cues. The simplest case of encoding the labeled pair (i, j) is to modify their weights so that

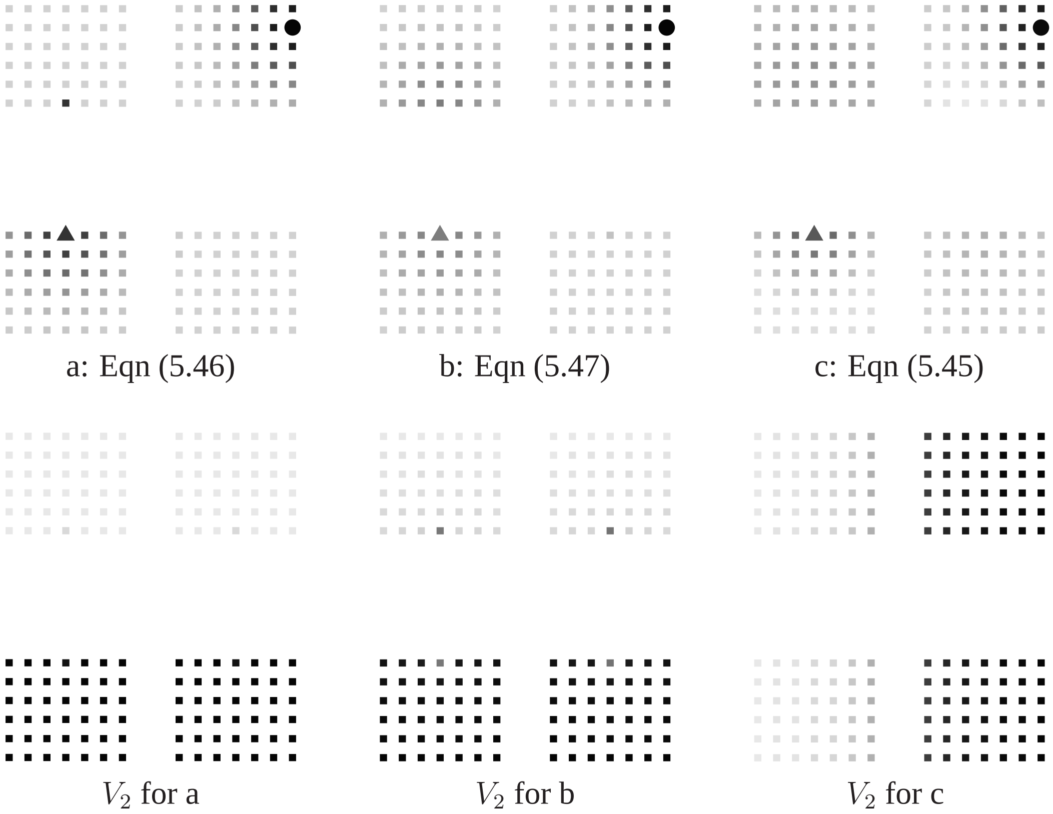
$$W_{ij} = W_{ji} := 1, \tag{5.46}$$

where an originally vanishingly small value increases to the maximum affinity. The influence of this change depends on the number of connections all nodes have. For example, if node i connects to 10 other nodes, this one more connection would matter little after being normalized by the total connections. Unlike minimum cuts, where a change in one link can change the global optimum completely, normalized cuts are insensitive to perturbation in weights. Another approach is to let the pre-assignment of i and j bridge their neighbours together:

$$W_{ik} = W_{ki} = W_{jk} = W_{kj} := \max(W_{ik}, W_{jk}),\ k \in \mathbb{V}. \tag{5.47}$$

Short-circuiting labeled nodes as well as their neighbours produces a similar result as the simple biased grouping in Fig 5.2. Their common problem is that only the labeled nodes expand their neighbourhoods significantly, which make them distinct from the rest unlabeled data. If we extend Eqn (5.47) to modify the weights *among* the neighbours of labeled points, we can overcome the discontinuity of the segmentation. That's what Eqn (5.45) does, and in a principled way.

The inherent flaw in our basic formulation is also evident in the undesirable results from even dense grouping cues. Though it is unclear for this point set



a: Eqn (5.46)

b: Eqn (5.47)

c: Eqn (5.45)

V_2 for a

V_2 for b

V_2 for c

Figure 5.4: Propagate partial grouping constraints. Row #1: QPQ values for one labeled point (▲) in Fig 5.2c and one unlabeled point (●). They are superimposed, with darker gray for larger values. a: Direct modification according to Eqn (5.46) only adds the other labeled point as its neighbour. b: direct modification according to Eqn (5.47) doubles the neighbourhood size for the labeled point. c: smoothed constraints allow the labeled point to have extensive correlations with all the nodes yet still maintaining fine differentiation toward its own neighbours and those of its labeled peer. The QPQ values on the unlabeled point change little. Row #2: the continuous optimum V_2 for normalized cuts.

what the best 4-class clustering is with either dense or sparse partial grouping cues, as shown in Fig 5.5, the labeled data points never stand out with smoothed constraints. In general we don't know how many classes there are and whether the partial grouping cues are sufficient; it is important to always use data coherence to smooth partial grouping constraints.

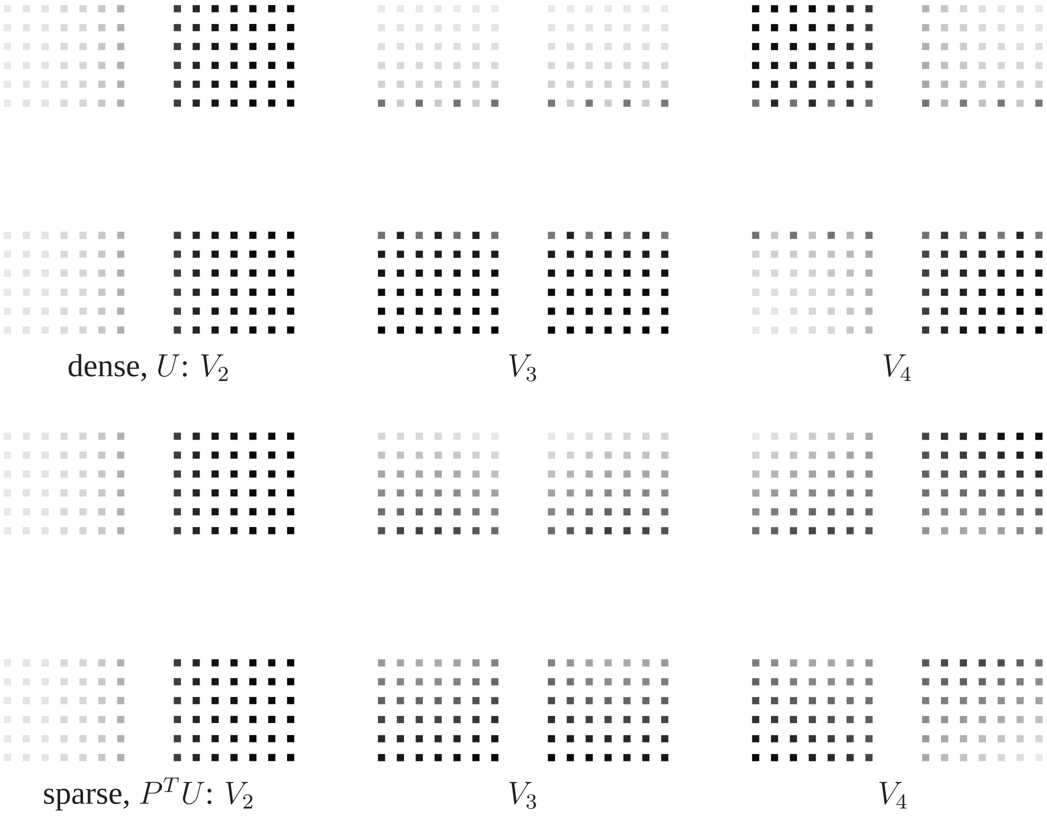


Figure 5.5: The importance of smoothing partial grouping constraints. Each row shows three leading eigenvectors. Row #1 are those for the dense grouping case in Fig 5.2b, with simple constraints U. Row #2 are those for the sparse grouping case in Fig 5.2c, with smoothed constraints $P^T U$. The first uniform eigenvectors (1_N) are omitted.

5.4 Experiments

We calculate pixel affinity using a Gaussian function on the maximum magnitude of intensity edges separating two pixels. $W(i, j)$ is low if i, j are on the opposite sides of a strong edge (Malik et al., 2001). Using this simple feature, we will demonstrate how simple extra-image knowledge can improve low-level segmentation and how smoothed partial grouping constraints make a difference.

In Fig 5.6, we derive partial groupings based on brightness values, e.g. the foreground is more likely to be lighter and the background is darker. We choose

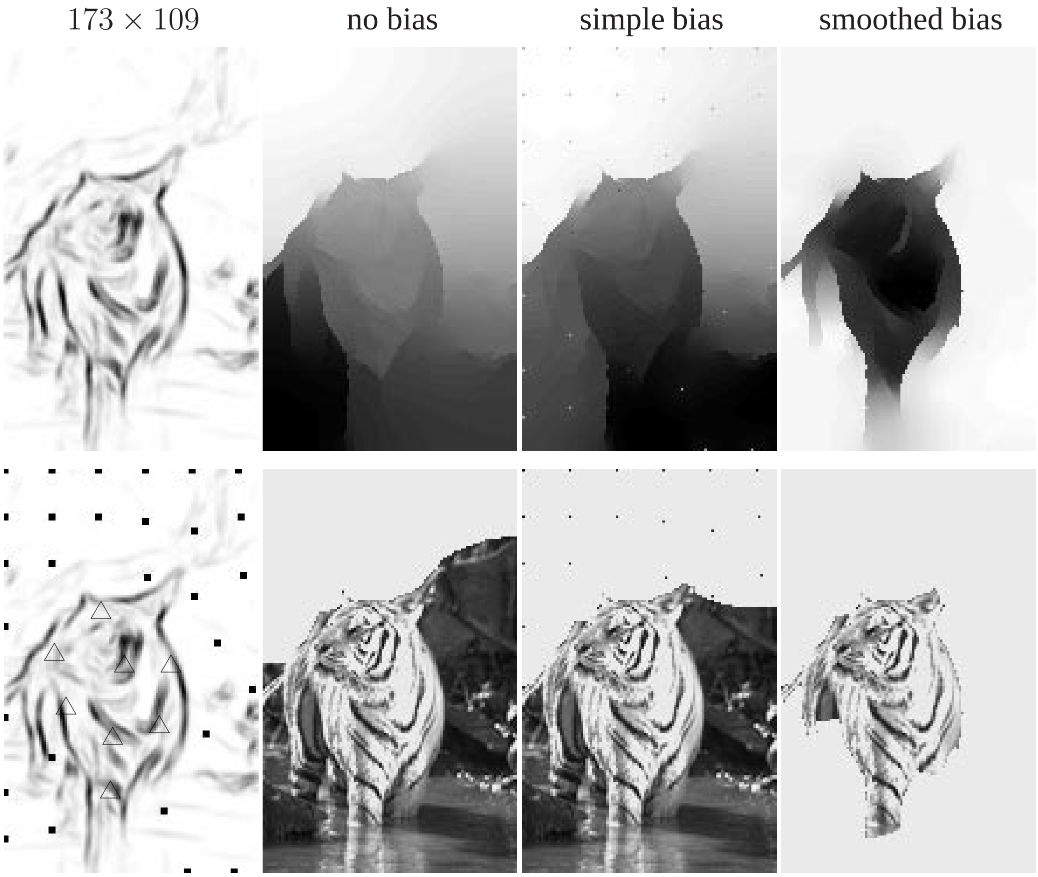


Figure 5.6: Segmentation with partial grouping from brightness. Column #1: edge magnitudes and biased nodes (29 pixels marked as ■, 8 pixels marked △) having extreme intensities. Columns #2-4: the second eigenvector and foreground images obtained with no constraints, simple constraints U and smoothed constraints P^TU respectively.

two thresholds to find the pixels at the two intensity extremes and then use morphological operations to further remove pixels appearing in the other set due to noise. As we have already seen in in Fig 5.2, with simple constraints, biased pixels stand out in segmentation, while with smoothed constraints, they bring their neighbours along and change the segmentation completely. This image has rich texture against a relatively simple background. Compared to segmentation using morphological operations on such images, our method can fill the holes caused by thresholding without losing thin structures or distorting region boundaries.

Partial grouping cues can also be derived from motion cues in a video sequence. In Fig 5.7, for every image, we compute its difference with two preceding images in a video sequence, threshold and then apply morphological operations to the difference image to create a mask for the foreground. Our constrained segmentation can effectively shrink it to the head in motion.

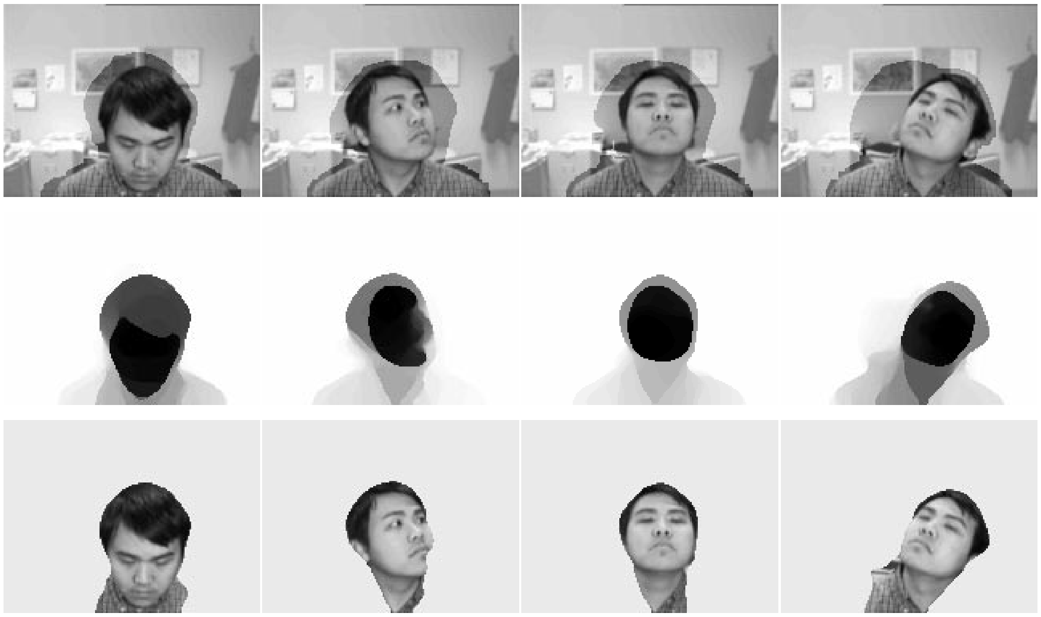


Figure 5.7: Segmentation with partial grouping from motion. A sequence of 120×160 images taken every 40 frames from a head tracking system. Row #1: images with peripheries masked out (contrast reduced) according to the difference with neighbouring images. The peripheries are pre-grouped together. Row #2: the second eigenvectors of constrained normalized cuts. Row #3: foreground images from discrete segmentation.

Partial grouping cues can come not only from low-level cues, but also from high-level expectation. For fashion pictures featuring a fashion model at the center, we choose the background to be: 4-pixel wide at left and right sides, and 7-pixel high at top and bottom sides. Fig 5.8 and Fig 5.9 show the results with and without such background knowledge. Notice that all eigenvectors of QPQ satisfy the constraints, and pixels at the four image sides always have similar values in the eigensolutions. Through these constraints, the large uniform background is never broken up in a segmentation, which focuses on the more interesting foreground-background separation or a division within the foreground itself.

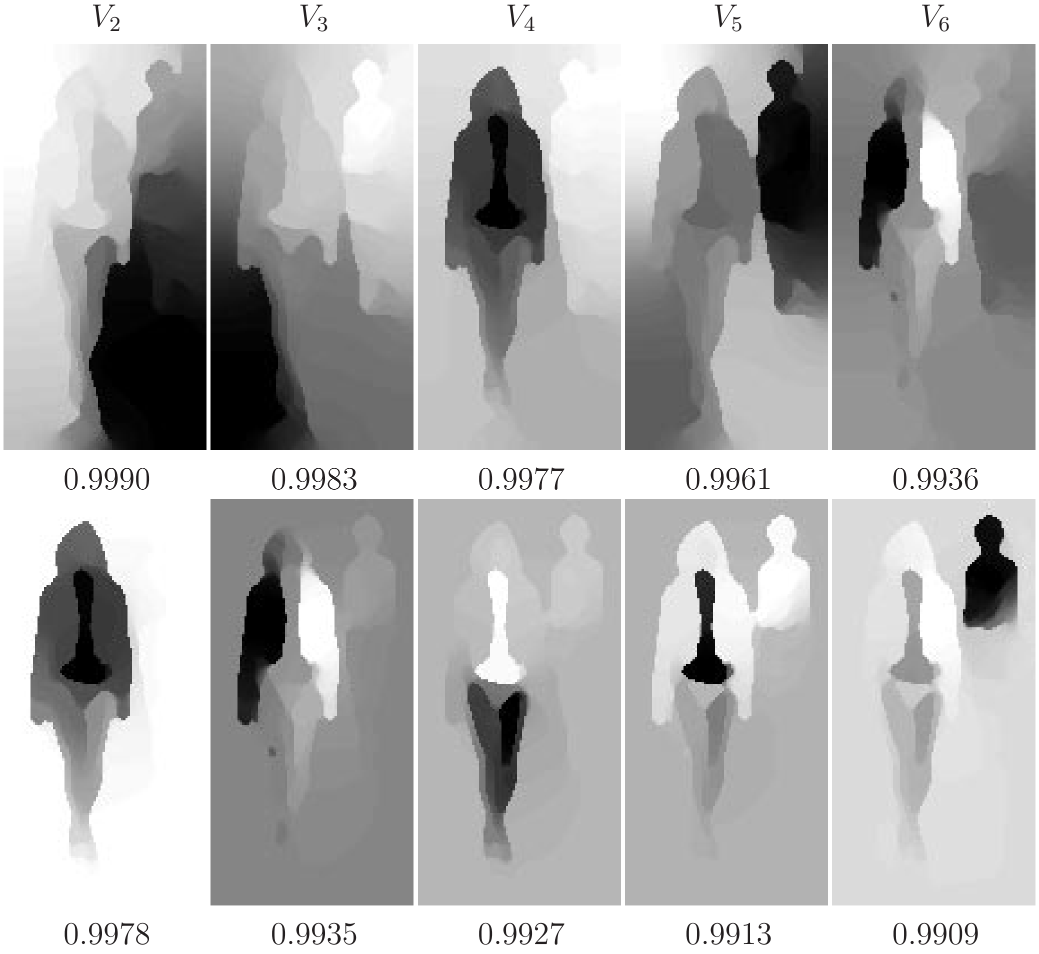


Figure 5.8: Segmentation with partial grouping from spatial attention. Image size: 180×90. Row #1-2: leading eigenvectors of unconstrained and constrained normalized cuts respectively. Uniform V_1's are omitted. Numbers are eigenvalues. It takes 27.2 and 19.7 seconds respectively to compute these eigenvectors in MATLAB on a PC with 1GHz CPU and 1GB memory.

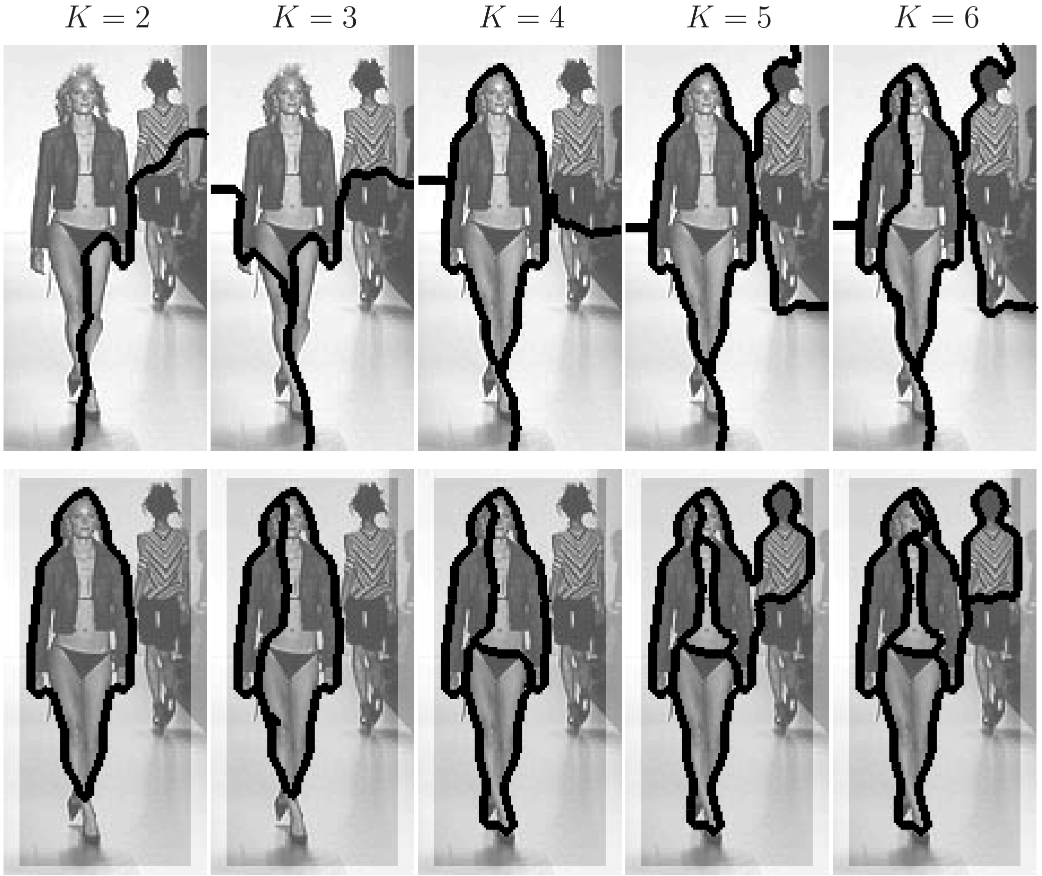


Figure 5.9: Multiclass segmentation derived from the eigenvectors shown in Fig 5.8. Row #1: unconstrained cuts. Row #2: constrained cuts.

Using the same spatial mask and the same set of parameters for computing pixel affinity, we apply our constrained normalized cuts to other fashion pictures and Berkeley image datasets (Martin et al., 2001). See Fig 5.10 and Fig 5.11. The number of classes K is chosen manually. When there is an object in the center of the image, such spatial priors always help a segmentation to pick out the object. If the prior is wrong, for example, when the background spatial mask touches the object of interest, e.g. the tip of shoes in the rightmost fashion picture, the final segmentation also removes the feet from the foreground. The extent of this detrimental effect depends on the connections of the constrained nodes, since partial grouping information is propagated to neighbouring nodes that they have large affinity with. Our formulation can neither spot nor correct mistakes in priors.

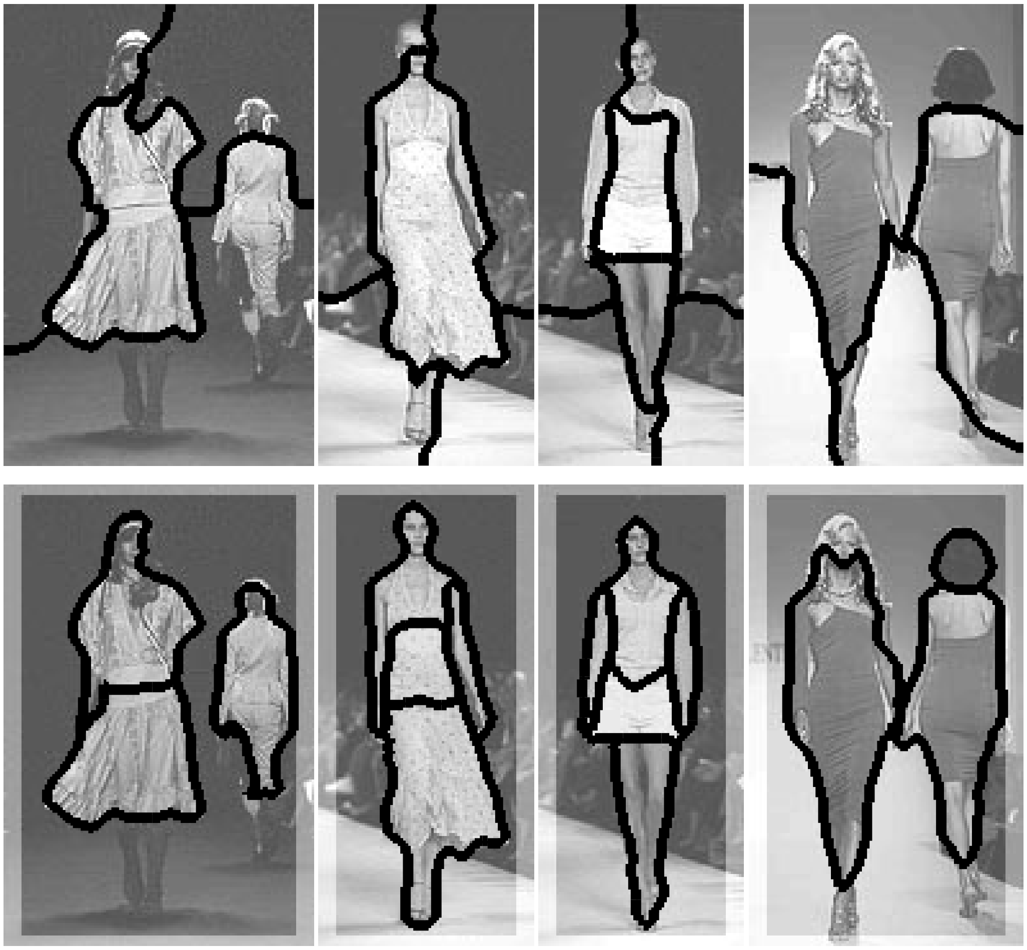


Figure 5.10: Segmentation without (row #1) and with (row #2) partial grouping at image boundaries, where contrast is reduced. Pictures are from New York Spring 2002 fashion shows.

Technically, Eqn (5.45) can be replaced by an up-to s-th order smoothness condition or a subset of it: $S_f = [P^0, P^1, ..., P^s]$. However, higher-order smoothness constraints propagate the partial grouping further at the cost of more com-

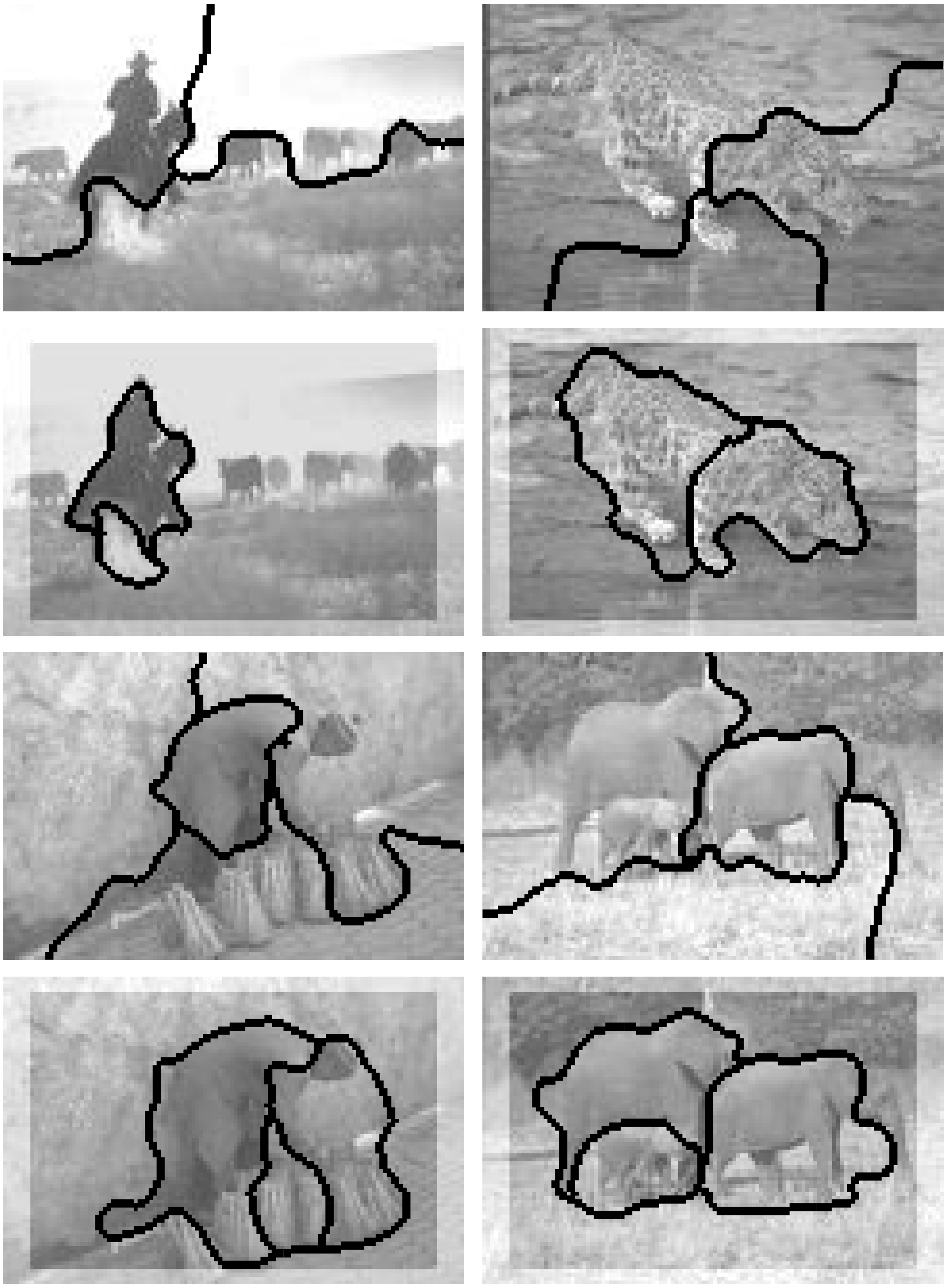


Figure 5.11: Segmentation without (row #1,3) and with (row #2,4) partial grouping at image boundaries, where contrast is reduced. Images are from Berkeley data sets.

putation. In our experiments, we also observe no significant improvement over $S_f = P$ in the eigensolutions.

5.5 Summary and Discussions

We developed a method that integrates both bottom-up and top-down information in a *single* grouping process. The former is based on low-level cues presented in the image itself, whereas the latter is based on partial grouping cues known *a priori*; the former defines the goodness of a segmentation, whereas the latter defines the feasibility of a segmentation. The two are unified in a constrained optimization problem. We showed that it is essential to propagate sparse partial grouping cues based on the coherence exhibited in the data themselves. In particular, we developed an efficient solution for such constrained normalized cuts and applied the method successfully to segmenting a wide range of real images.

Our work can be regarded as a small step toward bridging generative approaches and discriminative approaches for grouping. Generative models, including Markov random fields (Geman and Geman, 1984) and variational formulations (Blake and Zisserman, 1987; Mumford and Shah, 1985), can be naturally cast in a Bayesian framework, where data fidelity and model specificity are at equal footing. However, they are sensitive to model mismatches and are usually solved by Markov Chain Monte Carlo methods, which often find local optima with slow convergence.

Discriminative methods, for example graph approaches on image segmentation (Amir and Lindenbaum, 1996; Gdalyahu et al., 1998; Puzicha et al., 1998; Perona and Freeman, 1998; Sharon et al., 2000; Shi and Malik, 1997), achieve a global decision based on local pairwise relationships. These algorithms often have efficient computational solutions. These local pairwise comparisons can encode general grouping rules such as proximity and feature similarity. Promising segmentation results on a wide range of complex natural images were reported in (Malik et al., 2001). Such pairwise comparisons, however, often have difficulty in deriving reliable long-range grouping information.

Attempts have been made to find MRF solutions by graph partitioning algorithms (Greig et al., 1989; Ferrari et al., 1995; Boykov et al., 1998; Roy and Cox, 1998; Ishikawa and Geiger, 1998). In particular, sufficient and necessary conditions on the properties of energy functions that can be solved by *minimum* cuts have been proven in (Kolmogorov and Zabih, 2002b; Ishikawa, 2003). The work here shows that prior knowledge can be used to guide grouping for discriminative criteria such as normalized cuts (Shi and Malik, 1997), and that their global optima in the continuous domain can be solved algebraically with little extra cost.

Our work is also closely linked to the transduction problem, the goal of which is to complete the labeling of a partially labeled dataset (Joachims, 1999; Jaakkola et al., 1999; Nigam et al., 1999; Szummer and Jaakkola, 2001). If the labeled data set is rich enough to characterize both the structures of the data and the classification task, then using the induced classifier on the labeled set and interpolating it to the unlabeled set shall suffice, which is a supervised learning problem that has many efficient algorithms. However, usually the labeled set is small, so the problem becomes how to integrate the two types of information from both sets to reach a better solution. In (Joachims, 1999), the classification problem is formulated in the support vector machine (SVM) framework and labeled data are treated similarly to the rest except that their labels have been instantiated. In (Jaakkola et al., 1999), information about the labeled data is encoded in the prior distribution of the labeling and the goal is to find a projection of the best SVM discriminator onto the prior space. Through model averaging, partial labeling constraints are softly enforced. In (Nigam et al., 1999), class-dependent data generation models are assumed and the labeled data can be used to estimate the parameters involved in the models. This might be the most effective way to propagate priors. However, these generative models are often too simple to be realistic. In (Szummer and Jaakkola, 2001), the class-dependent probability models are hidden in the pairwise affinity matrix of all the data points. Again, the labeled set is used to estimate the class-dependent label generation process.

Though our work was initially motivated by the gap between discriminative and generative approaches, we are aware of other works that put similar constraints into clustering algorithms such as K-means (Wagstaff et al., 2000; Wagstaff et al., 2001). Two types of constraints, *must-link* and *cannot-link*, are considered. An earlier version of our work (Yu and Shi, 2001a) also considered cannot-link constraints, that is, two nodes cannot assume the same label. It involves approximation and is not included here.

Our work is distinct from all these methods in two aspects. Rather than instantiating the labels or the constraints on labeled data points, we use them to regulate the form of a segmentation. We gave an intuitive computational account for the need of constraint propagation and provided a principled way to implement it. Secondly, we can solve near-global optima of our formulation, whereas most other works can only guarantee local optimality.

Our experimental results on image segmentation demonstrate that simple grouping bias can approach figure-ground segregation without knowing what the object is. Our spatial priors effectively take advantage of the asymmetry between figure and ground (Amir and Lindenbaum, 1998a). In other words, since the outcome of a grouping depends on global configuration, figure-ground segregation can be obtained not only by enhancing the saliency of object structures, but also by suppressing background structures, the latter of which is often easier than the former. Our next step is to explore the integration of more complicated priors in order to segment out only objects known *a priori*.

Chapter 6

Object Segmentation

The problem we want to solve in this chapter is illustrated in Fig 6.1: partition an image into foreground and background, with objects of interest in the foreground and unknown clutter in the background.

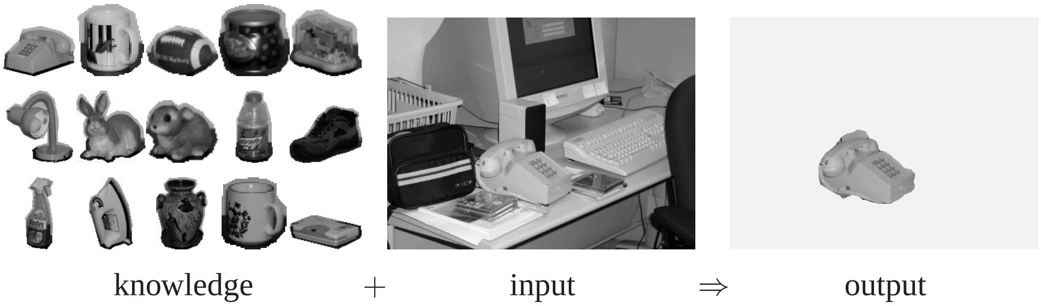


Figure 6.1: The goal of object segmentation.

In most object segmentation formulations, only one type of object is considered and object knowledge is employed to overcome data noise. It is often used when the object of interest is known to be present and some initial estimation of the size and location can be obtained. For example, in the deformable template approach (Xu et al., 2000), a deformable prototype is used with a deformation space modeled from training data. Some well-known applications are: detecting the eye and mouth (Yuille et al., 1989), tracking shapes in motion (Blake and Isard, 1998), and segmenting anatomical parts in medical images (McInerney and

Terzopoulos, 1996).

An alternative to deformable templates for object segmentation is proposed in (Borenstein and Ullman, 2002). Instead of a globally constrained template, object knowledge is represented using pairs of image fragments and their figure-ground labeling from a training set. The problem of segmentation becomes one of finding an optimal cover on a test image with a set of training fragments whose appearances match the image and whose labeling patterns are locally compatible.

This exemplar-based approach is appealing for its flexible representation of objects. However, the authors only show results on low-resolution (40×30) images, each of which has an object occupying the center, with little background. There are a few problems not easily addressed in their framework:

1. Hallucination. If falsely detected fragments happen to align well locally, there is no way to prevent a wrong segmentation. This occurs very often when the background has significant clutter.

2. Imprecision. Since the segmentation of a test image comes from a collection of local segmentations of *training* images, details of object boundaries in the test image are inevitably lost.

3. Restricted to single object. Their energy function is only suitable for one object present in an image, not for multiple objects from either the same or different classes. It is not trivial to relate cover scores from different objects such that partial covers from multiple objects are always inferior to a whole cover for one true object.

All these top-down object segmentation approaches require image data to conform to object models, whether encoded in templates or fragments. Here, adopting image patches as a representation, we propose a concurrent segmentation and recognition system that also addresses the above mentioned problems.

Our basic idea is that image segmentation should take into account both low-level feature saliency and high-level object familiarity (Peterson, 1994). With the guidance of object knowledge, segmentation will not get lost in image noise

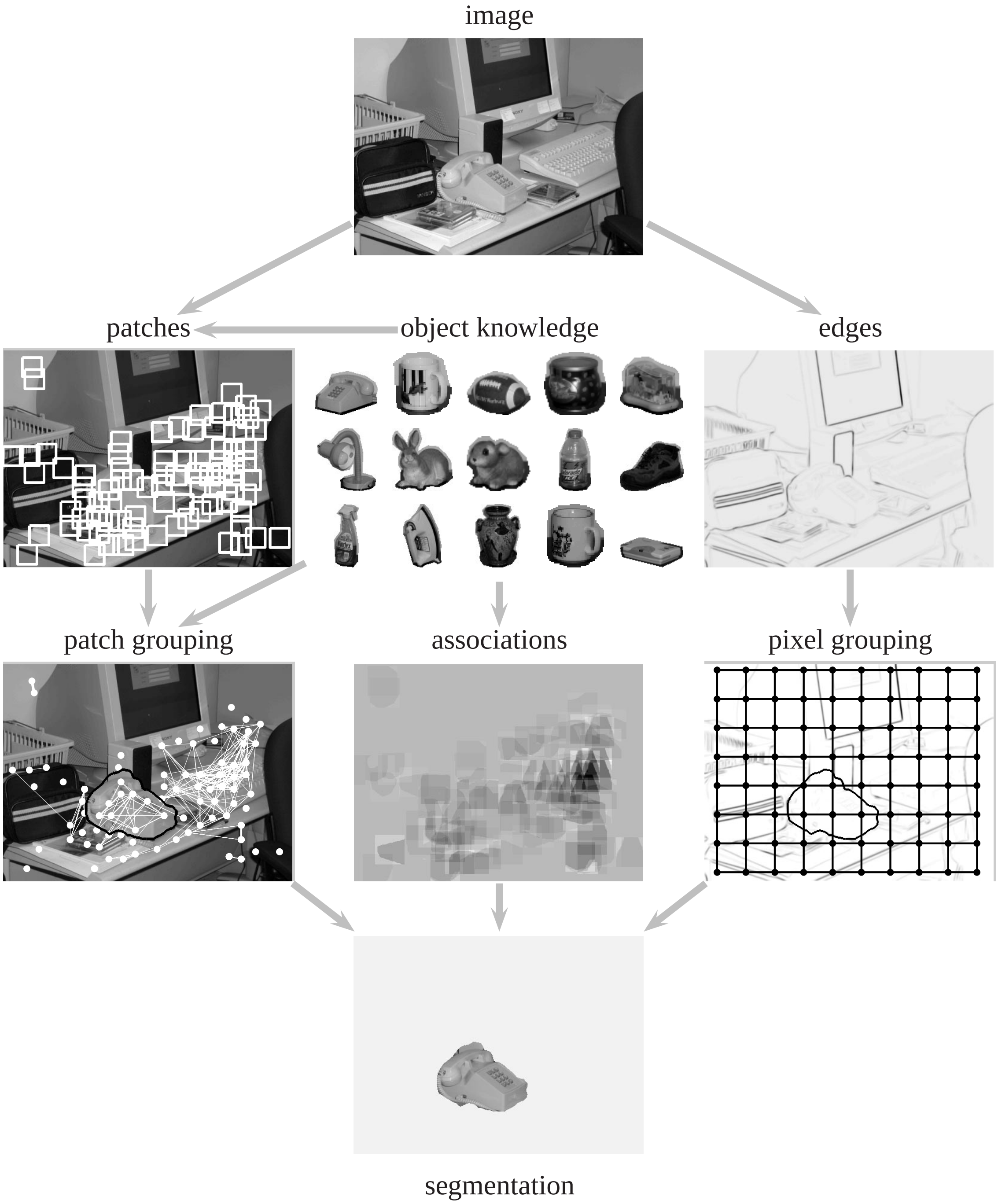


Figure 6.2: Our approach. A set of patches are identified with object parts in a training set. Patches of consistent spatial configurations are sorted in patch grouping. Every object part hypothesis is also associated with a local segmentation of pixels. Here we show an overlay of such local segmentations. Dark for figure, white for background, gray for the non-committed. At the low-level, edges are first detected. Pixels of similar intensities are sorted in pixel grouping. Object segmentation is obtained by coupling the patch and pixel grouping in their solution space, where the consistency endowed by the patch-pixel associations is enforced.

and background clutter. With the verification of low-level feature saliency, we prevent the hallucination of falsely detected object parts standing out from their surroundings.

We formulate our method in a graph-theoretic framework. As illustrated in Fig 6.2, we first detect patches and edges, and then we build two relational graphs for patch and pixel grouping. They share the same representation and grouping criterion, except that the former has patches as nodes and hypothesis compatibility as affinity between two nodes, while the latter has pixels as nodes and feature similarity as affinity between two nodes. We optimize a combined grouping criterion in a reduced solution space where patch-pixel correspondence is encoded. These constraints facilitate figure-ground segregation to produce object patches and their pixels in the foreground group, and the rest in the backgroud group. Built upon our earlier work on constrained cuts (Yu and Shi, 2001a), we can solve near-global optimal solutions efficiently.

6.1 Integration Model

In this section, we focus on the integration problem, i.e., given pixel grouping cues, patch grouping cues, pixel and patch correspondence cues, how do we integrate them for image segmentation? We will illustrate what these cues represent here, and defer the issue of obtaining these cues until the next section.

6.1.1 Representation: Affinity and Indicators

In a graph-theoretic approach, a graph is specified by nodes, edges and their associated weights. Nodes represent the elements to be grouped. Every pair of nodes are connected by an edge, with a weight describing the likelihood of the two elements belonging together. We assume this weight is nonnegative and symmetric. The pairwise relationships between N nodes and M other nodes can be summarized in an $N \times M$ matrix, called *affinity matrix*. When they are the same set of nodes, the affinity matrix becomes symmetric.

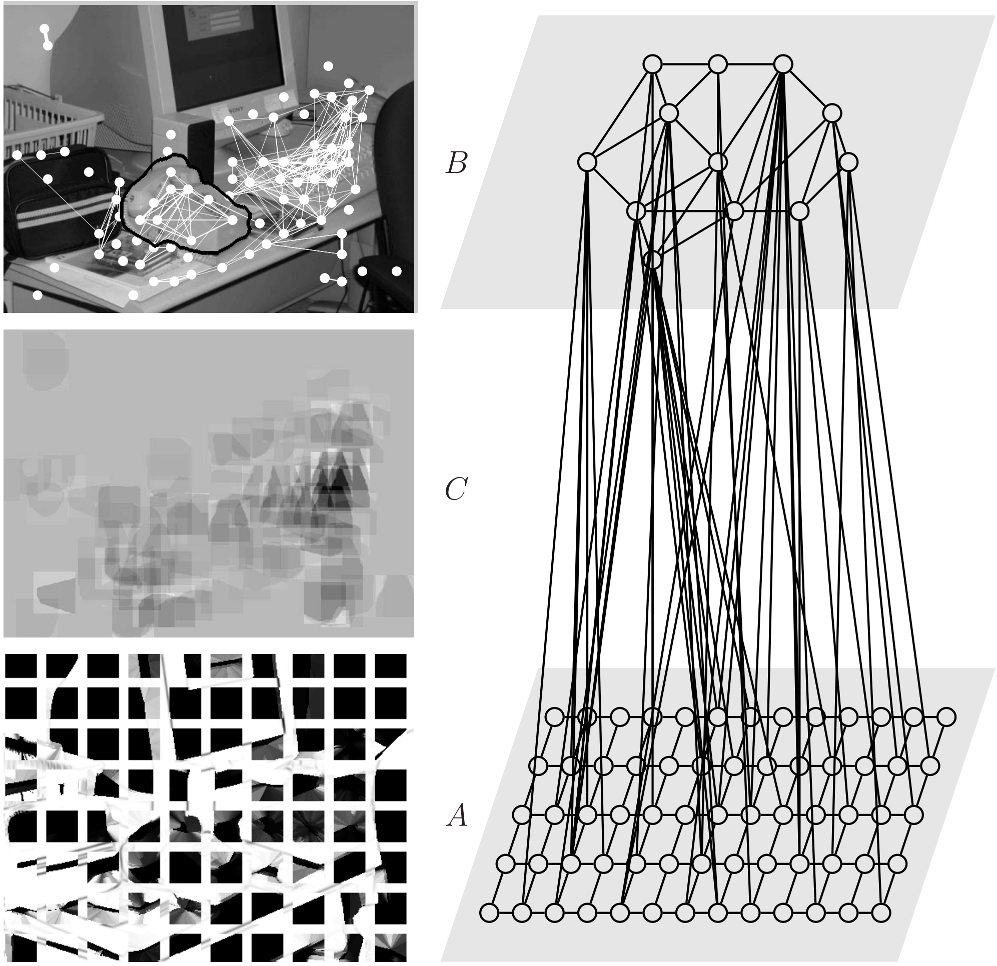


Figure 6.3: Graph representation. Each pixel and patch in the image is a node in the graph. A: connections between pixels. They are derived from an edge map, with low values for pixels on the opposite sides of an edge. The affinities of a set of regularly spaced pixel nodes to their neighbours are superimposed on the edge map of the image. Darker gray for larger values. B: connections between patches. They are derived from object models, with low values for patches misaligned spatially. Thicker lines for larger affinity. C: connections between pixels and patches. Each pixel has different associations to different patches. Here is a summation of the associations to all patches. Darker gray for larger association to some object parts.

We formulate our grouping problem by graph $\mathbb{G} = \{\mathbb{V}, \mathbb{U}; A, B, C\}$, where node set $\mathbb{V} = \{1, \ldots, N\}$ denotes a total of N pixels, node set $\mathbb{U} = \{N + 1, \ldots, N + M\}$ denotes a total of M patches, affinity matrix $A_{N\times N}$ denotes pixel similarity, affinity matrix $B_{M\times M}$ denotes patch compatibility, affinity matrix $C_{N\times M}$ denotes pixel-patch associations. See Fig 6.3.

Object segmentation now becomes a node partitioning problem. Given node set $\mathbb{V}$, let $\Gamma_{\mathbb{V}}^K = \{\mathbb{V}_1, \ldots, \mathbb{V}_K\}$ denote a division of $\mathbb{V}$ into K disjoint sets: $\mathbb{V} = \cup_{l=1}^K \mathbb{V}_l$, $\mathbb{V}_k \cap \mathbb{V}_l = \varnothing$, $k \neq l$. Our goal is to find consistent $\Gamma_{\mathbb{V}}^K$ and $\Gamma_{\mathbb{U}}^K$ so that $\forall l$, $\mathbb{V}_l$ and $\mathbb{U}_l$ contain either corresponding object pixels and patches, or background pixels and patches. The ordering is inconsequential.

We introduce *probabilistic group indicators* to represent a partition. Let $X = [X_1, \ldots, X_K]$, where $X_l(i) = \Pr(i \in \mathbb{V}_l)$. Similarly, we define $Y = [Y_1, \ldots, Y_K]$ for patch grouping $\Gamma_{\mathbb{U}}^K$.

6.1.2 Criterion: Goodness of Grouping

Partitioning within $\mathbb{V}$ or $\mathbb{U}$ itself is a basic grouping problem, for which we adopt the normalized cuts criterion (Yu and Shi, 2003). Take pixel grouping as an example. We maximize the average within-group connections defined by:

$$\begin{aligned}
\varepsilon(\Gamma_{\mathbb{V}}^K; A) &= \frac{1}{K}\sum_{l=1}^{K} \mathrm{linkratio}(\mathbb{V}_l, \mathbb{V}_l; A) && (6.1)\\
\mathrm{linkratio}(\mathbb{V}_l, \mathbb{V}_l; A) &= \frac{\mathrm{links}(\mathbb{V}_l, \mathbb{V}_l; A)}{\mathrm{degree}(\mathbb{V}_l; A)} && (6.2)\\
\mathrm{degree}(\mathbb{V}_l; A) &= \mathrm{links}(\mathbb{V}_l, \mathbb{V}; A) && (6.3)\\
\mathrm{links}(\mathbb{P}, \mathbb{Q}; A) &= \sum_{i\in\mathbb{P}, j\in\mathbb{Q}} A(i,j). && (6.4)
\end{aligned}$$

Due to the normalization, normalized cuts also minimize the average between-group connections at the same time. See details in (Yu and Shi, 2003). Likewise, for patch grouping, we desire a partitioning that maximizes $\varepsilon(\Gamma_{\mathbb{U}}^K; B)$.

Given $\Gamma_{\mathbb{V}}^K$ and $\Gamma_{\mathbb{U}}^K$, our joint criterion $\bar{\varepsilon}$ takes both individual goodness and relative importance into account:

$$\bar{\varepsilon}(\Gamma_{\mathbb{V}}^K, \Gamma_{\mathbb{U}}^K; A, B) = \frac{1}{K}\sum_{l=1}^{K} \text{linkratio}(\mathbb{V}_l, \mathbb{V}_l; A) \cdot \frac{\text{degree}(\mathbb{V}_l; A)}{\text{degree}(\mathbb{V}_l; A) + \text{degree}(\mathbb{U}_l; B)} \\ + \frac{1}{K}\sum_{l=1}^{K} \text{linkratio}(\mathbb{U}_l, \mathbb{U}_l; B) \cdot \frac{\text{degree}(\mathbb{U}_l; B)}{\text{degree}(\mathbb{V}_l; A) + \text{degree}(\mathbb{U}_l; B)}. \tag{6.5}$$

The combination coefficients of the connection ratios make sure that weights with a larger unit are weighed more in the criterion. The above definition, however, is different from a direct convex combination of $\varepsilon(\Gamma_{\mathbb{V}}^K; A)$ and $\varepsilon(\Gamma_{\mathbb{U}}^K; B)$. Here we weigh each pair, $\mathbb{V}_l$ and its counterpart $\mathbb{U}_l$, separately. In fact, we have:

$$\bar{\varepsilon}(\Gamma_{\mathbb{V}}^K, \Gamma_{\mathbb{U}}^K; A, B) = \varepsilon\left(\Gamma_{\mathbb{V}\cup\mathbb{U}}^K; \begin{bmatrix} A & \\ & B \end{bmatrix}\right), \tag{6.6}$$

i.e., cuts using the joint criterion $\bar{\varepsilon}$ are equivalent to the normalized cuts on the joint graph with the joint weight matrix.

We introduce further notation. For any nonnegative matrix A, let D_A denote its degree matrix. It is a diagonal matrix with $D_A(i,i) = \sum_j A(i,j), \forall i$. We rewrite Eqn (6.5) using group indicators X and Y, assuming that they are binary:

$$\bar{\varepsilon}(X, Y; A, B) = \frac{1}{K}\sum_{l=1}^{K} \frac{Z_l^T W Z_l}{Z_l^T D_W Z_l}, \tag{6.7}$$

$$Z = \begin{bmatrix} X \\ Y \end{bmatrix}, \qquad W = \begin{bmatrix} A & \\ & B \end{bmatrix}, \qquad D_W = \begin{bmatrix} D_A & \\ & D_B \end{bmatrix}. \tag{6.8}$$

We use this formula to extend the definition of $\bar{\varepsilon}$ in Eqn (6.5) to the real domain so that it gives a meaningful value when X and Y are probabilistic.

6.1.3 Criterion: Feasibility of Grouping

The objective value $\bar{\varepsilon}$ cannot guarantee the consistency between pixel grouping and patch grouping. That is, its optimum might not be interpretable in an object segmentation. Ideally, patches in $\mathbb{U}_l$ have their pixels in $\mathbb{V}_l$, and vice versa. When such grouping correspondence is enforced, we have a smaller but meaningful set of segmentations to look at. Among these feasible solutions, the one yielding the best $\bar{\varepsilon}$ is the desired grouping.

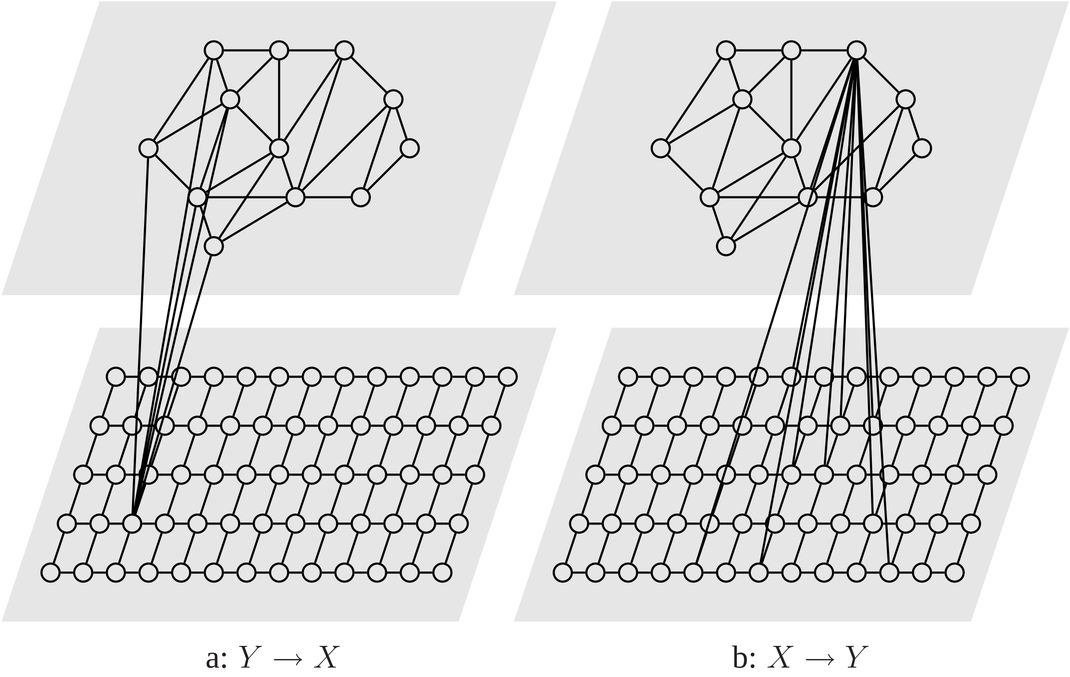


Figure 6.4: Consistency between high-level and low-level grouping. The foreground probabilities of nodes at one level are influenced by their neighbours at the other. a: when a set of patches all support the presence of an object, their common pixel is likely to be foreground; when in conflict, patches compete to claim the common pixel with their association strengths. b: for pixels within a coherent region, they unanimously bring their common patch into the counterpart patch group. Otherwise, the patch is drawn to the patch group that pairs with the pixel group to which its dominant pixels belong.

If the spatial configuration of some patches are consistent with their roles as object parts, then they belong in the same patch group, e.g. $\mathbb{U}_k$. This decision so far has nothing to do with the low-level pixel grouping; it is entirely based on high-level object knowledge. However, the implication of this patch grouping on pixels is clear. Pixels of these patches are more likely to be in $\mathbb{V}_k$, regardless of their dissimilarity in low-level features. Conversely, if a set of pixels with similar features belong to $\mathbb{V}_k$ in pixel grouping, patches claiming these pixels are more likely to be grouped in $\mathbb{U}_k$, regardless of their incompatible spatial displacement. See Fig 6.4. This often happens for falsely detected parts, which occupy areas without a boundary delineation in terms of low-level features, thereby we can easily pull these patches into the background.

Such double competition between high-level and low-level grouping can be described by constraints on the two group indicators X and Y through the association affinity C. Detailed in the next section, C reflects expected local segmentations, object vs non-object, within detected patches. That is, the value $C(i, p)$ is large if pixel i is likely to be an object pixel of patch p. We first encode the between-patch competition by re-weighting among patches:

$$\bar{C}(i, p) = C(i, p) \cdot \frac{C(i, p)}{\max_q C(i, q)}. \tag{6.9}$$

For pixel i, its association with patch p does not change if it is the strongest among all the patches; otherwise, $C(i, p)$ gets damped by its proportion to the maximum weight so that weak connections become even weaker.

After the non-maximum suppression among patches, we consider between-pixel competition by normalizing weights among pixels:

$$Y = D_{\bar{C}}^{-1} \bar{C} X. \tag{6.10}$$

This equation links the probabilities for nodes in one set to the other. For example, given the foreground probability of every pixel, the foreground probability of a patch is the weighted average of those of its member pixels. If the majority of these pixels belong to $\mathbb{V}_k$, then this patch as well as any other patch claiming most of these pixels is probably in $\mathbb{U}_k$. Eqn (6.10) can be rewritten as

$$LZ = 0, \quad L = [D_{\bar{C}}^{-1} \bar{C}, -I]. \tag{6.11}$$

where I is an identity matrix and L is assumed full rank.

6.1.4 Solution: Constrained Optimization

Putting the goodness and feasibility of grouping together, we have a constrained optimization problem:

$$Z^* = \arg\max \bar{\varepsilon}(Z; W), \quad \text{subject to} \quad LZ = 0. \tag{6.12}$$

Our low-level pixel grouping and high-level patch grouping are coupled in their solution space through pixel-patch interactions. We have a modular computational framework, yet it is not at all feedforward.

Note that our formulation is not the same as maximizing $\varepsilon(\Gamma_{\mathbb{V}}^K; A + \bar{B})$, which is a simple addition of two grouping processes, with the patch affinity B converted into an equivalent pixel affinity matrix $\bar{B} = (D_{\bar{C}}^{-1}\bar{C})^T B (D_{\bar{C}}^{-1}\bar{C})$ using the constraint in Eqn (6.10).

Eqn (6.12) is in the form of a constrained optimization problem that we have already considered in (Yu and Shi, 2001a) and its near-global optima can be solved efficiently. There, simple partial pixel grouping cues were fixed *a priori*. Here, these cues are specified with respect to a set of patches, the groups to which they belong are not known themselves. The uncertainty in the pixel and patch memberships is removed after a joint optimization process that respects both the grouping criterion and the consistency constraints.

To summarize, below is an overview of our algorithm.

```
1:  Detect edges.
2:  Evaluate pixel feature similarity $A$.
3:  Detect patches.
4:  Evaluate patch compatibility $B$.
5:  Evaluate pixel-patch association $C$.
6:  Form constraint matrix $L$.
7:  Solve constrained normalized cuts on $W$ and $L$ by eigendecomposition.
8:  Discretize the eigenvectors for a final segmentation.
```

6.2 Implementations

An image is first convolved with oriented filters to extract edge magnitudes. Pixel affinity A is evaluated using a Gaussian function (with standard deviation σ_e) on the maximum magnitude of edges crossing the line connecting two pixels. $A(i, j)$ is low if i, j are on the opposite sides of a strong edge (Malik et al., 2001).

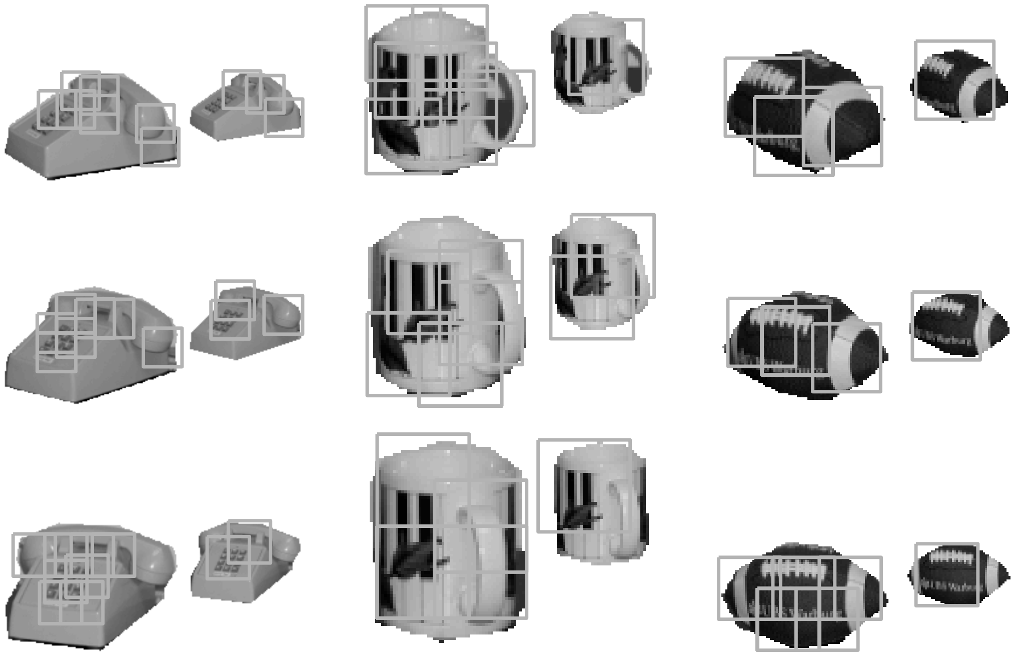


Figure 6.5: Samples of object parts in a training set. There are 15 objects in total (Fig 6.1). Each object is represented using 18 training images. These training images were acquired at 2 scales and 9 viewing directions that were 20° apart from each other.

For patches, we use the nearest neighbour object part detector described in (Mahamud, 2002), but without the final verification of whole object hypotheses. Shown in Fig 6.5, parts are represented by exemplars obtained from a few angles and scales. Local color, intensity, and orientation histograms are computed as features. Based on an optimal distance measure d learned from a training set in order to maximize the discrimination among objects, patch p is labeled with the nearest neighbour p' with score $d(p, p')$. See Fig 6.6. There could be multiple patches detected at the same location, corresponding to multiple object part hypotheses of the same local area in the image.

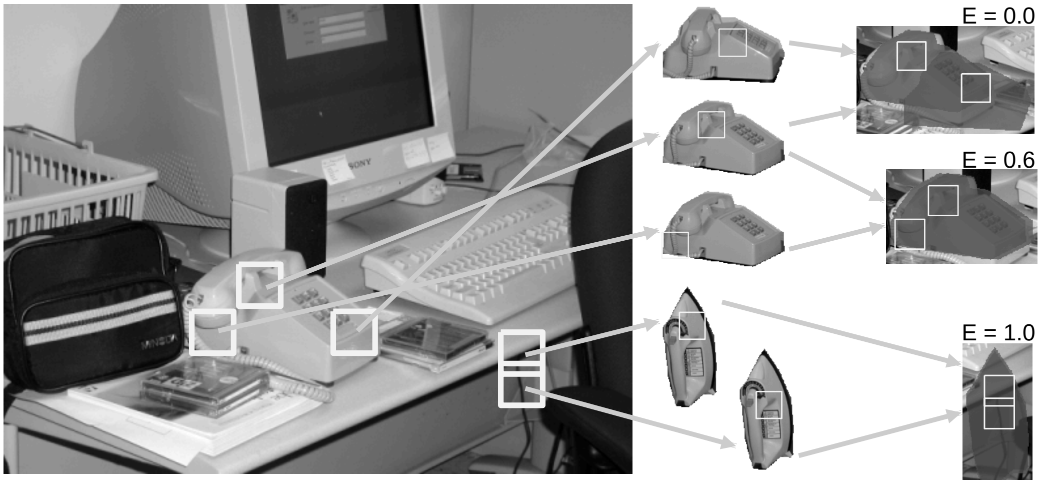


Figure 6.6: Patches have high affinity if their corresponding object silhouettes overlap.

We measure the quality of detected patches in two terms. One is the credibility of each individual patch denoted in a diagonal matrix B_O:

$$B_O(p,p) = \exp\left(-\frac{1}{2\sigma_p^2}\left[\frac{d(p,p') - d_{min}}{d_{max} - d_{min}}\right]^2\right), \tag{6.13}$$

where d_{max} and d_{min} are the minimum and maximum d values of all patches in image f. The more similar patch p is to the object part p' in the training set, the better the credibility. The other is the compatibility of a patch with nearby patches. Let $S(p',p)$ be the binary object silhouette of the training image to which part p' belong, registered to the location p in image f. Two patches p and q are consistent if $S(p',p)$ and $S(q',q)$ overlap well. This measure increases with the distance between p and q:

$$\begin{aligned} B_S(p,q) = \exp&\left(-\frac{1}{2\sigma_s^2}\left[1 - \frac{||S(p',p) \wedge S(q',q)||_1}{||S(p',p) \vee S(q',q)||_1}\right]^2\right) \\ &\cdot \left[1 - \exp\left(-\frac{1}{2\sigma_d^2} \cdot -\frac{||\underline{p} - \underline{q}||_2^2}{r(p) \cdot r(q)}\right)\right], \end{aligned} \tag{6.14}$$

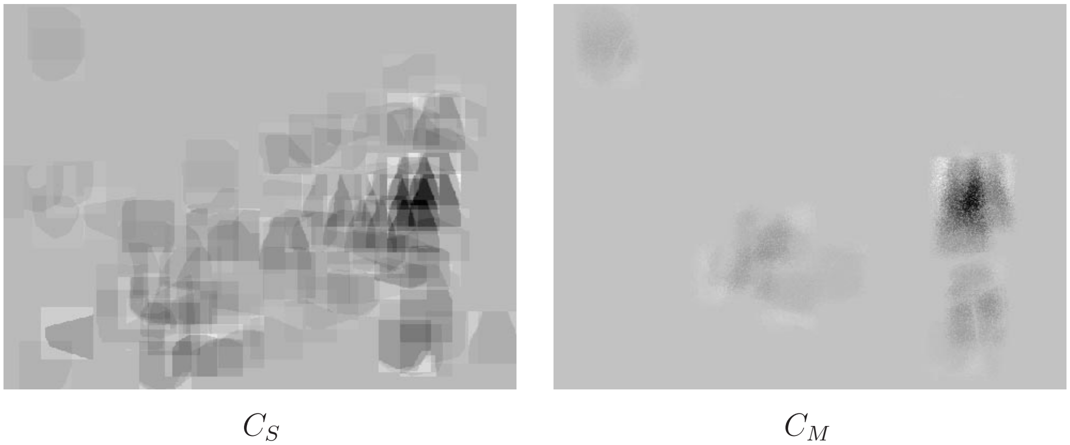
where $\wedge$ and $\vee$ are the logical and/or operators, $||\cdot||_k$ is L_k-norm, $r(p)$ is the radius of patch p, and $\underline{p}$ is the coordinates of the center of patch p in the image. In particular, $B_S(p,q)=0$ if $\underline{p}=\underline{q}$. Multiplying a patch's own credibility and its compatibility with others together, we obtain patch affinity as:

$$B_M = B_O^T \cdot B_S \cdot B_O. \tag{6.15}$$

This value is high when a patch is not only similar to an object part in the training set, but also spatially aligned with other detected patches for the same object. As a result, an isolated falsely detected patch has very low patch affinity. Finally, to balance the units between pixel and patch grouping in our joint grouping criterion, we scale the patch affinity with a constant so that the total degrees match between pixel and patch graphs:

$$B = \frac{1_N^T A 1_N}{1_M^T B_M 1_M} \cdot B_M, \tag{6.16}$$

where 1_d denotes the $d \times 1$ vector of all ones.



C_S C_M

Figure 6.7: Pixel-patch associations. C_S is a projection of local segmentations from object models. After diffusion with both pixel and patch affinity, C_M is refined and propagated. Each pixel has different associations to different patches. Here we show the summation of its weights to all patches. Dark gray for foreground, white for background.

The object silhouette $S(p',p)$ also projects an expected local segmentation of pixels at the detected patch location. We denote it by matrix C_S, each column of which has N pixels, taking the corresponding values of $S(p', p)$ within a window 25% larger than the patch itself: $+1$ for object pixels and -1 for background pixels. Taking an average of C_S based on affinity with neighbouring pixels and patches, we refine this initial estimation by:

$$C_M = A \cdot C_S \cdot B. \tag{6.17}$$

After such diffusion on the signed representation for projected local segmentations, the associations between pixels and object patches are weakened near false boundaries expected at an edge-less region (Fig 6.7). The final pixel-patch associations are established between patches and their object pixels only:

$$C = C_M \odot (C_M > 0), \tag{6.18}$$

where $\odot$ denotes the element-wise product.

In total, we have four parameters for the Gaussian functions used to evaluate the affinity measures, and they are fixed for all test images: $\sigma_e = 0.02$, $\sigma_p = 0.33$, $\sigma_s = 0.08$, $\sigma_d = 0.17$.

6.3 Experiments

In Fig 6.8, we compare grouping in three conditions: 1) low-level pixel grouping only, where $\varepsilon(\Gamma_{\mathbb{V}}^2; A)$ is maximized; 2) a scheme in which object detection is used to narrow down a region of interest (ROI) which contains several candidate objects, and then $\varepsilon(\Gamma_{\mathbb{V}}^2; A)$ is maximized for these pixels; 3) a joint pixel-patch grouping. Low-level grouping alone segments out a large region, which, although coherent in the low-level features, is irrelevant to our objects of interest. With focus of attention, low-level grouping picks out a small area, which, although well separated from its surroundings, corresponds to a region of falsely detected object patches. Only with the guidance of patch grouping, the object of interest, despite its weak contrast at the boundaries, pops out from the rest of the clutter.

Our method of enforcing constraints in the solution space is often contrasted with a straightforward alternative where the patch grouping interacts with pixel

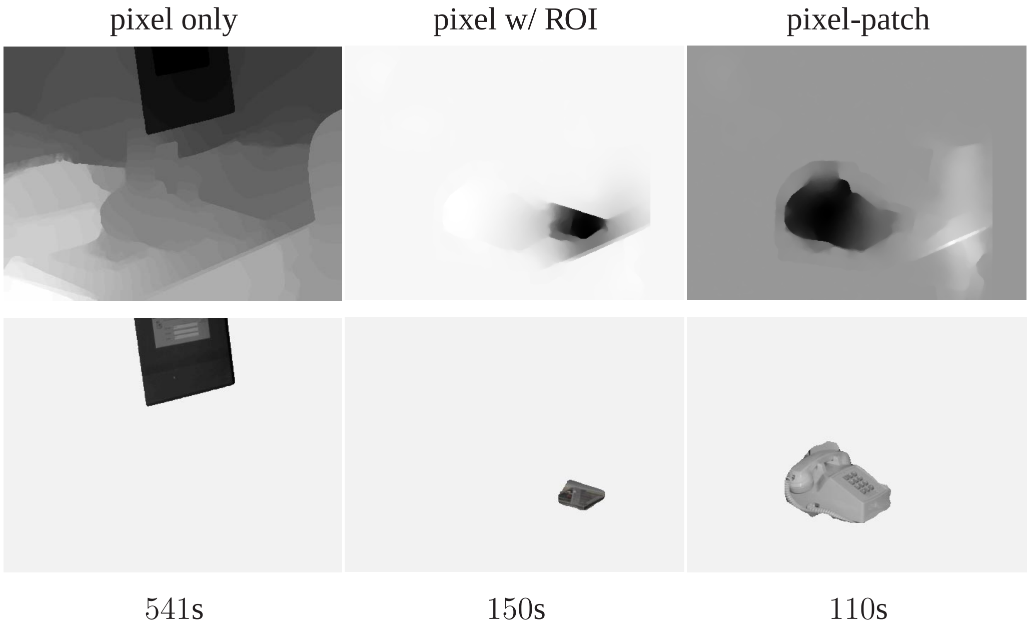


Figure 6.8: Comparison of pixel grouping, focused pixel grouping and pixel-patch grouping. Row #1: optimal eigenvectors. Row #2: segmentations. The MATLAB running times are given with 1GHz CPU and 1GB memory.

grouping directly in the weight matrix (Yu et al., 2002):

$$W = \begin{bmatrix} A & C \\ C^T & B \end{bmatrix}, \tag{6.19}$$

and we then simply seek normalized cuts on this weight matrix. There are two reasons why this alternative is inferior. First, weights of C as interactions between two processes are of a different nature from those between pixels or between patches and there is no obvious way to account for this with a reasonable joint criterion. Therefore, although we can still run normalized cuts on the resulting graph, we do not know what we are actually optimizing. Secondly, such pixel-patch associations can cause hallucination regardless of pixel grouping (Fig 6.9). Of course, the degree of hallucination depends on the relative weights among A, B, C and whether the pixel-patch associations agree with the low-level grouping. In general, however, such associations emphasize pixels linked to detected patches, causing these pixels to stand out, a phenomenon we have already observed in biased grouping (Yu and Shi, 2001a).

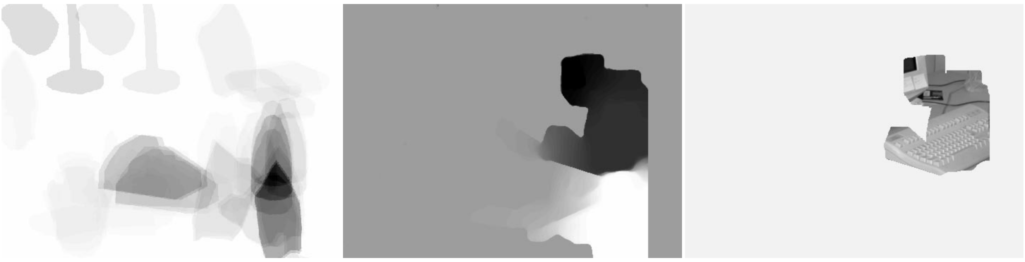

a: density of patches b: eigenvector c: segmentation

Figure 6.9: Direct pixel-patch associations as in Eqn (6.19) provide wrong bias with falsely detected patches. a: overlay of object silhouettes weighted by the degrees of the patches D_B. The false detections happen to align well so that they overshadow the real object parts. b,c: spectral clustering results if these pixel-patch associations are directly included in the affinity matrix. Pixels associated with patches are heavily biased, which cause the hallucination of a non-existent object, ignoring evidence in pixel grouping.

We apply our method to over 400 test images. Fig 6.10 to Fig 6.12 are a sample of the results. There are about 800 patches detected in total, however, as seen in the patch density images, only a few (about $30 - 50$) patches have significant connections to other patches. Most patches either score low in matching object parts in the training images, or have no nearby patches detected for the same object at a similar scale and pose.

When the object has well-defined boundaries in the test image (e.g. Fig 6.10 #1,2,5), then it can be segmented despite occlusion, imprecision in measurements of part location, orientation and scale. However, since an object is represented with a few patches, which, although selected to maximumly discriminate between the 15 objects, do not necessarily cover the whole object or can all be detected. As a result, when an object has strong interior edges (e.g. Fig 6.10 #3), only the area inside its strong edges are segmented. When an object has very weak contrast at its boundaries (e.g. Fig 6.10 #4, Fig 6.11 #1, 2), it is often segmented with a piece of the background that shares similar low-level features. When none of the object

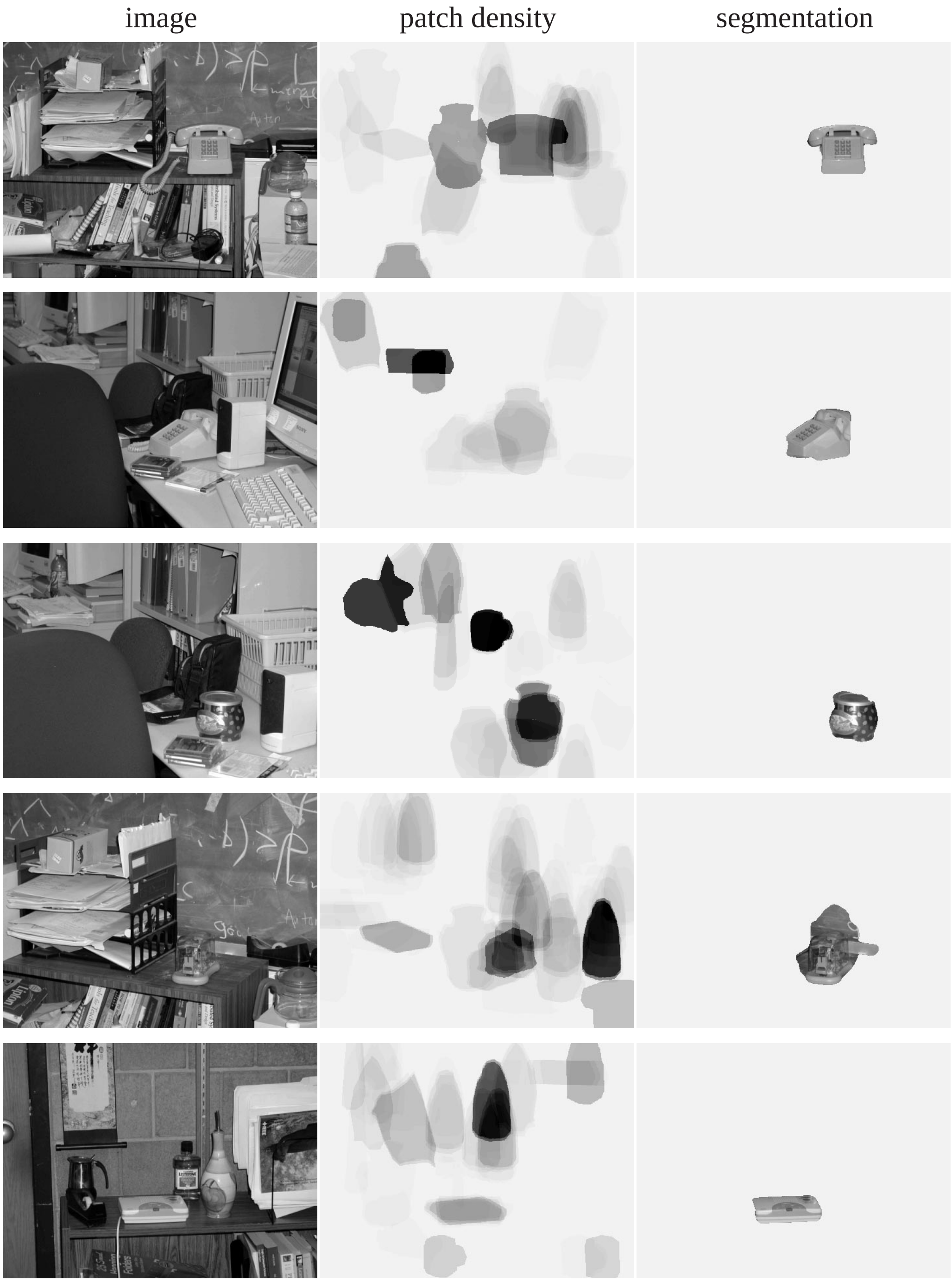


Figure 6.10: 2-class object segmentation. Patch density is defined as in Fig 6.9.

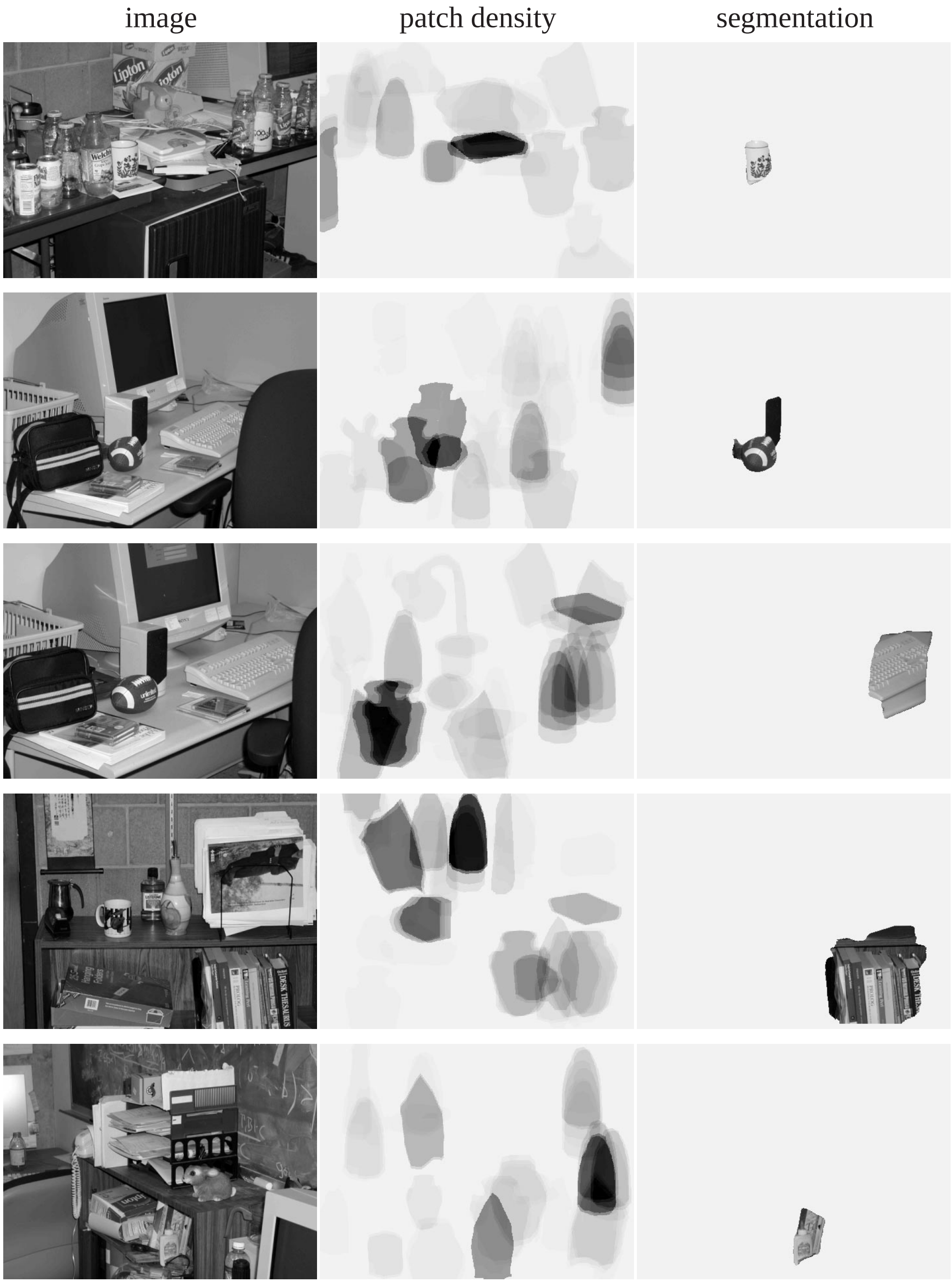


Figure 6.11: 2-class object segmentation. Same convention as Fig 6.10.

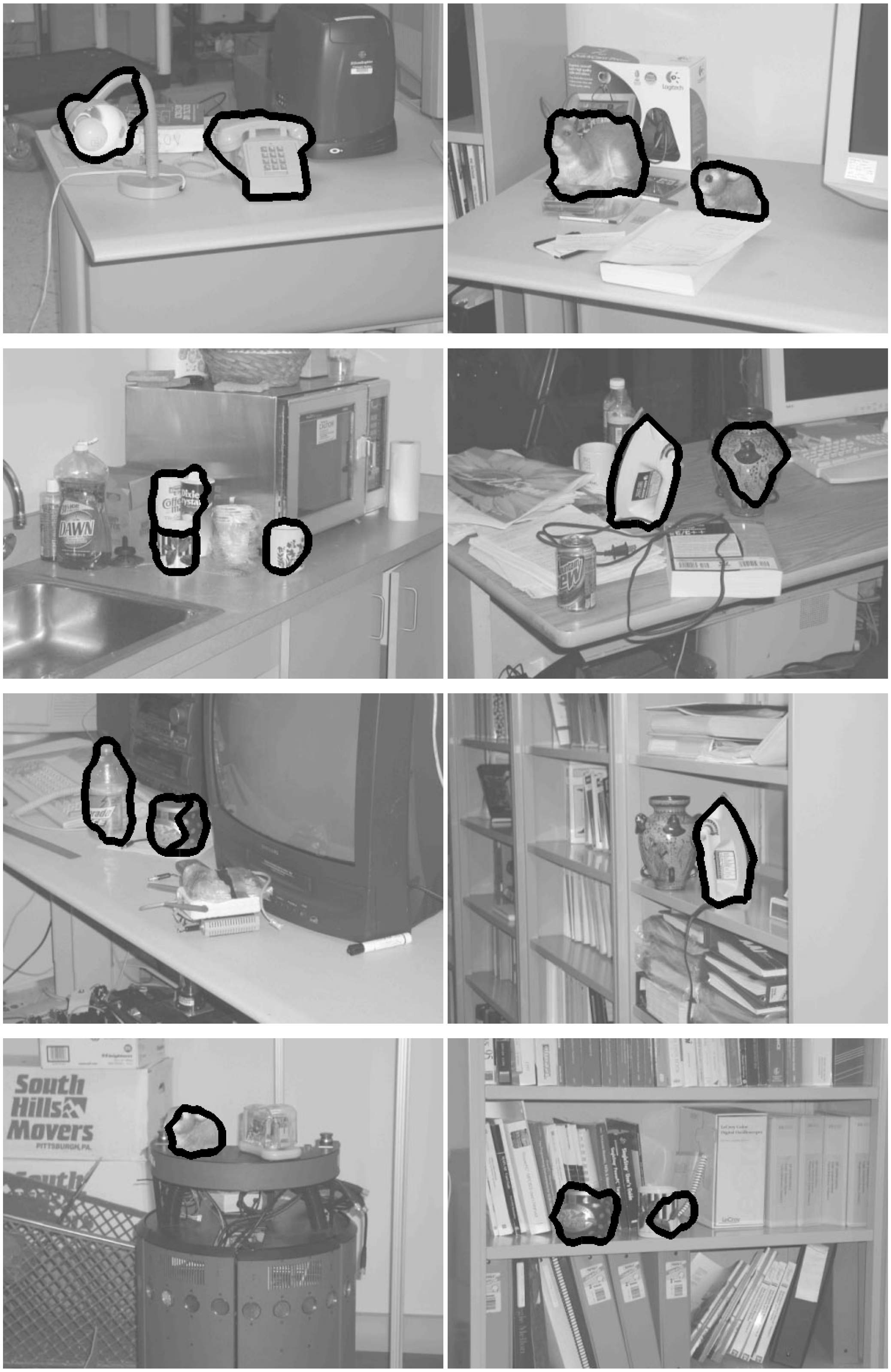
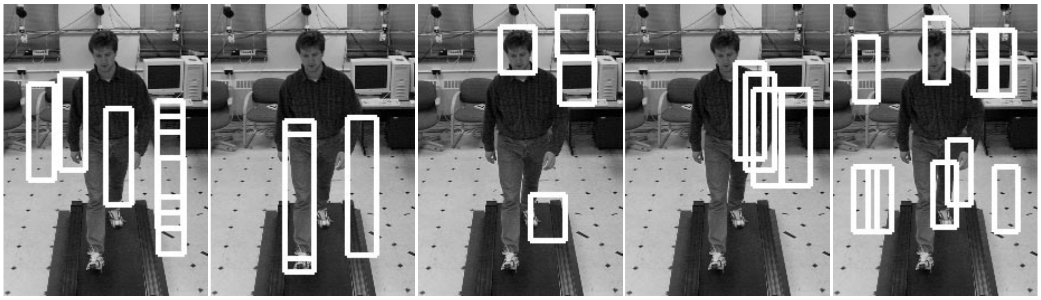


Figure 6.12: Multiple object segmentation results.

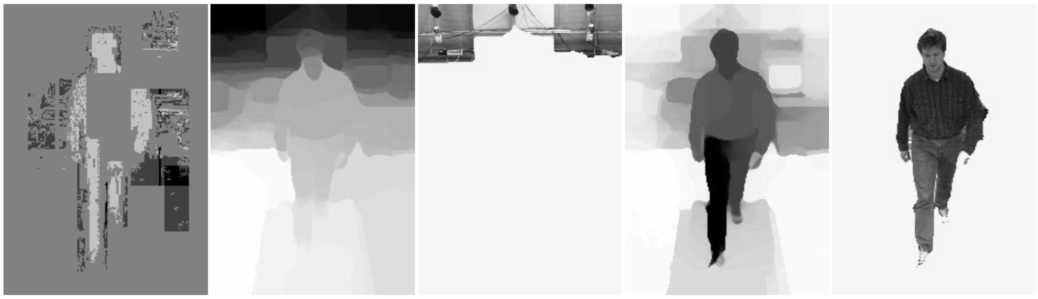
parts are detected (Fig 6.11 #5), a region resembling another object (the iron in Fig 6.1) becomes the foreground object. Sometimes, the object parts are not well localized (e.g Fig 6.11 #4), thus the object boundaries are obscured, causing the object to merge into background. These problems are also evident with multiple objects. See Fig 6.12.



Right arm: 7 Right leg: 3 Head: 4 Left arm: 4 Left leg: 9

Figure 6.13: Human body part detection. 27 patches are detected, each labeled by one of the five part detectors for arms, legs and head. False positives cannot be validated on two grounds: they are not distinct from their surroundings or incompatible with nearby parts.

We apply our method to articulated objects, i.e. human body segmentation in a single image (Yu et al., 2002). My colleague Ralph Gross developed an algorithm to detect human body parts. He manually labeled five body parts (both arms, both legs and the head) of a person walking on a treadmill in all 32 images of a complete gait cycle. Using the magnitude thresholded edge orientations in the hand-labeled boxes as features, a linear Fisher classifier (Fukunaga, 1990) is trained for each body part. In order to account for the appearance changes of the limbs through the gait cycle, two separate models are used for each arm and each leg, bringing the total number of models to 9. Each individual classifier is trained to discriminate between the body part and a random image patch. The classifiers are iteratively re-trained using false positives until the optimal performance is reached over the training set. In addition, linear color-based classifiers are learned for each body part to perform figure-ground discrimination at the pixel level. Alternatively a general model of human appearance based on filter responses as in (Sidenbladh and Black, 2001) could be used. Fig 6.13 shows detected parts for a test image.



pixel-patch C_S segmentation alone: 68s segmentation-recognition: 58s

Figure 6.14: Spectral segmentations for the 261×183 image in Fig 6.13. The results are shown with both the second eigenvector of normalized cuts and the optimal discrete segmentation based on the eigensolution.

For articulated objects, there is no simple formula for patch affinity. We show some preliminary results in Fig 6.14. Though the pixel-patch affinity matrix C_S, derived from the color classifier, is neither precise nor complete, and the edges are weak at many object boundaries, the two processes complement each other in our pixel-patch grouping system and output a reasonably good object segmentation.

6.4 Summary

We have developed a joint optimization model to integrate detected edges and object parts to produce an object segmentation. Our results show that it does not hallucinate object boundaries like most top-down object segmentation approaches, nor does it get lost in irrelevant regions of rich features as do most low-level image segmentation approaches. Imprecision in patch detection and poor contrast of edges are tolerated to a certain degree.

However, the experimental results are far from being satisfactory for a number of reasons. First, detecting and localizing enough patches is not always possible. To be detected and to be indicative of a particular object, an image patch must have distinctive low-level features. For the purpose of segmentation, we also desire these patches to be sensitive to the pose of the object. There might not be enough such patches available for the object of interest, especially when they are small (e.g. mugs), symmetrical across rotations (e.g. lamps), or have repeating texture patterns (e.g. vases). Secondly, the computed patch affinity is not always reliable. Without a flexible representation for the geometrical configurations between object patches, the errors in patch locations and poses can be easily amplified in the object silhouettes, leading to wrong affinity values. Finally, without a model of object shapes, textures of the object and weak contrasts at object boundaries can interfere with the segmentation.

To handle object shapes, we could include another process – contour grouping into object segmentation (Yu and Shi, 2001b). With the guidance of object models, we might also eliminate the major problem in low-level contour grouping: random continuation of edgels (Williams and Jacobs, 1997). How to get a good estimation of grouping correspondence also warrants further research.

On the other hand, our formulation can be considered an integration framework for node grouping and hyper-edge grouping. Instead of viewing patches as independent nodes, we can regard them as hyper-edges defined on basic elements – pixel nodes. The interaction matrix describes their incidence relationships. This provides a way to include high-order relationships into a grouping framework that only deals with pairwise relationships. It has already been noted that pairwise relationships are not enough to describe grouping constraints. For example, we need to describe cues or hypotheses that depend on other cues. Our work could potentially provide such a representation.

Chapter 7

Conclusions

Why does the human vision system perceive the world so effortlessly and instantly? Our answer to this question is that our perceptual organization pops out sensory information that is ecologically relevant. Such sensory information can be roughly divided into two categories: that due to *novelty* and that due to *familiarity*. Popout by novelty helps us to immediately detect potential dangers in the world, while popout by familiarity prompts us for an immediate reaction to friends and foes. Both novelty and familiarity can be driven by low-level and high-level cues. For example, red among blue is novel; an exotic object among commonplace objects is also novel. Likewise, familiarity can be as general as good curve continuation, or as specific as a particular face.

These two perceptual phenomena, popout by novelty and popout by familiarity, are what we studied computationally in this thesis. We considered novelty cues triggered in a bottom-up mechanism, where pixels with large local feature contrast with neighbouring pixels become one outlier group that demands further processing. We also considered familiarity cues triggered in a top-down mechanism, where only the objects we know *a priori* form the foreground that demands further object identification.

In our computational models of perceptual popout, we took a discriminative approach rather than a generative approach. For generative approaches, organizing objects into groups is equivalent to understanding all aspects of the objects.

For example, segmenting out an orange would require knowing that oranges are usually yellow, round, and lightly textured. Such details are often not needed, e.g. if the orange is embedded in a background with nothing yellow or round. What discriminative approaches require, on the other hand, are exactly such cues from local comparisons, indicating whether two visual elements are in the same group. That's why discriminative approaches are more suitable for modeling popout.

Our ideas are in contrast with the traditional computational models of perceptual organization, where only feature similarity is emphasized and the whole process is thought of as a precursor independent of specific object knowledge. We support the view that perceptual processing is an interactive process that involves perceptual organization and object recognition simultaneously. Built within the framework of spectral graph theory, we were able to achieve what most other interactive processing methods fall short of one way or another:

1. We clearly stated a criterion gauging the goodness of perceptual organization, which provide us a clear understanding of the structure of the solutions without being distracted by any solution techniques.

2. We developed an efficient and principled algorithm for finding the solutions to our criterion. Our solutions in the continuous domain are global optima, which guarantee that the discrete solutions obtained as the closest ones to the continuous optima are near global optima to our criterion.

3. We demonstrated the use of our approach on segmenting a wide range of real images.

Specifically, we developed a set of simple yet realistic interactive processing models for real image segmentation. As demonstrated on hundreds of real images, we were able to understand more about the computational principles underlying perceptual popout; we were able to unify grouping cues, figure-ground cues and depth cues in one grouping framework; finally, we were able to achieve object segmentation with cues derived from spatial and object attention (Fig 7.1).

We achieved these contributions by empowering current spectral graph models of perceptual organization with a richer representation of grouping cues, a

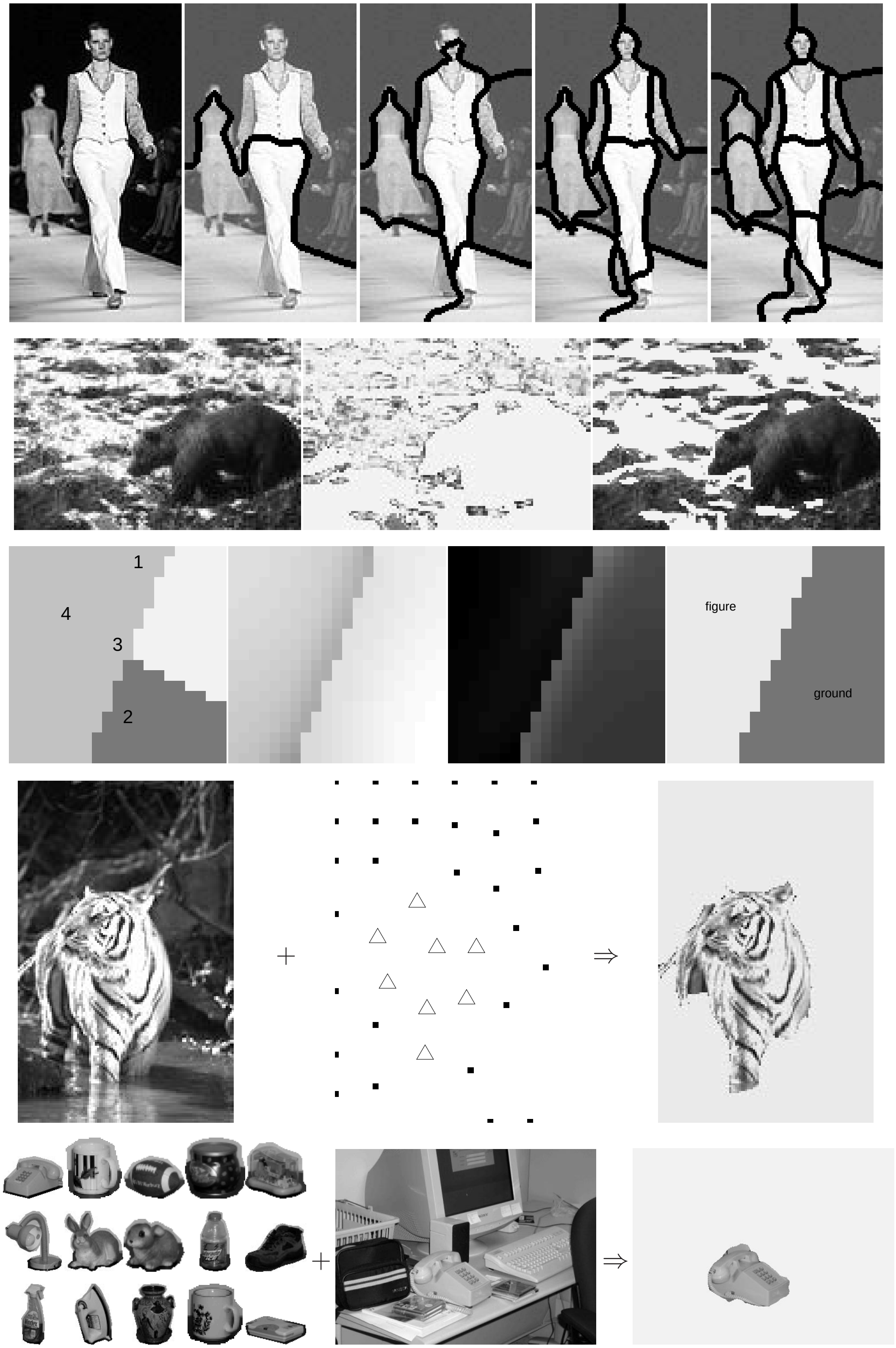


Figure 7.1: What we achieved about image segmentation in this thesis. One row for each computational model: multiclass, repulsion, depth, bias and object segmentation.

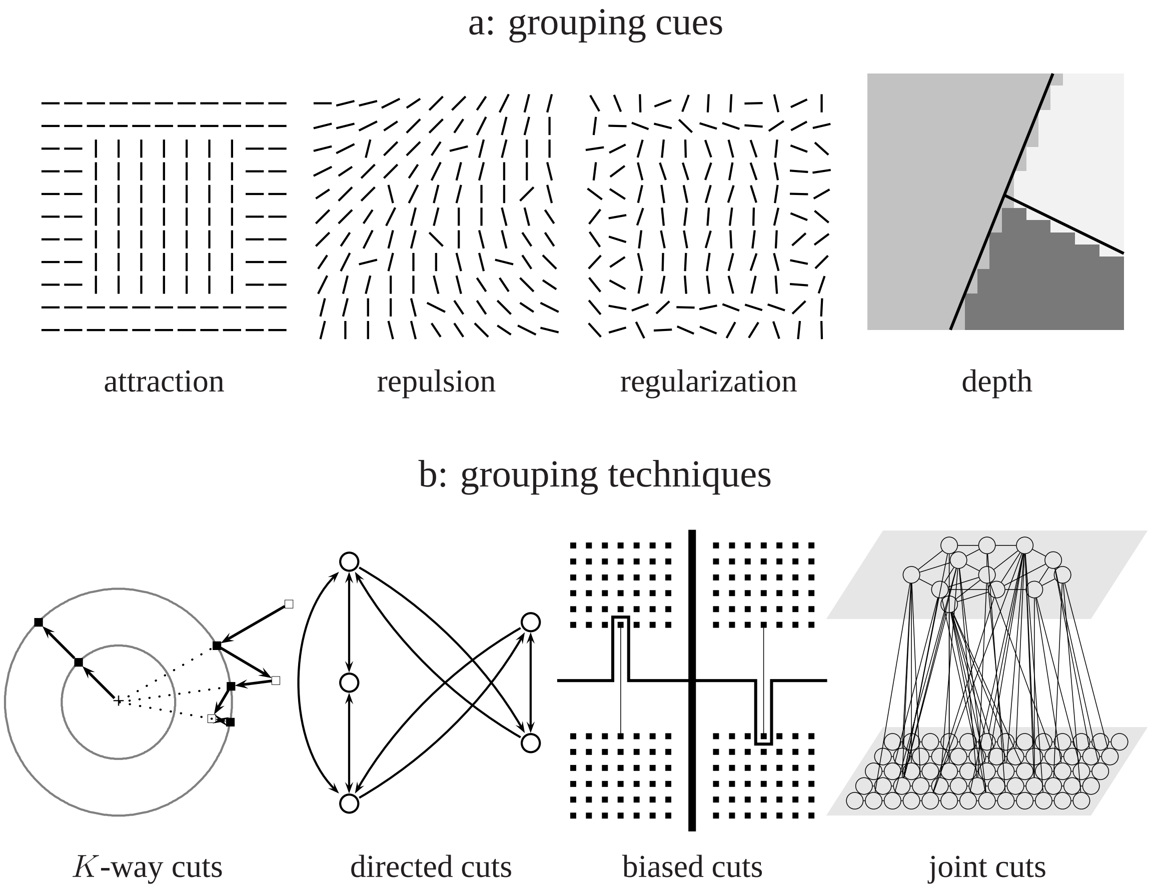


Figure 7.2: Computational tools developed in this thesis. They can be divided along two lines of development: grouping cues and grouping techniques, although some are inter-related. a: We expanded the repertoire of grouping cues from attraction to repulsion (Chapter 3) and depth (Chapter 4), and in the framework of attraction and repulsion, we can enhance the confidence of grouping cues by regularization (Chapter 4). b: We extended the grouping techniques to multiclass (Chapter 2), ordered partitioning (Chapter 4), constrained partitioning (Chapter 5) and joint partitioning (Chapter 6).

grouping criterion that accounts for all cues simultaneously, and a computational solution that finds near-global optima efficiently (Fig 7.2).

There are a few important issues worth further exploration.

1. Automatic selection of the number of classes for image segmentation. How many groups are there? This problem is seemingly rather subjective. However, we find that analyzing how our objective value changes over the number of classes can give a set of candidate answers.

2. A model-based view of spectral clustering. Recently a model-based view (Kamvar et al., 2002) was given to other clustering algorithms such as K-means and hierarchical agglomerative algorithms. Likewise, a related criterion in graph partitioning, *minimum cuts*, has an intimate relationship with Markov random fields (Boykov et al., 1999; Kolmogorov and Zabih, 2002a) and thus a generative interpretation. This leaves spectral clustering alone as a technique unexplained in terms of generative models. Such a model-based view can provide insights on predicting the behavior of an algorithm and also reveal its connections to other methods.

3. A criterion for comparing two segmentations. Automatic evaluation of image segmentation is desired as the demand to process huge image datasets increases. Some heuristics, for example, the normalized correlation between two labeling matrices in a formula such as $|A \cap B|/|A \cup B|$, have been used. Such a pixel-to-pixel comparison does not take the structural similarity of two segmentations into account. Because of that, even if we have a "gold standard" such as manual segmentation, the numbers we come up with may still be meaningless.

4. Closing a feedback loop. Though we have provided a method for integrating top-down information, we have not developed a mechanism for monitoring mistakes caused by bad priors or bad data. On the other hand, it has been demonstrated that the human vision system can ignore incongruent depth cues so long as the 2D projections make up a familiar object (Bulthoff et al., 1998). Incorporating a feedback mechanism would probably involve the comparison of alternative grouping results in order to correct the adverse influence of bad cues.

5. Object representation. Despite the flexibility of exemplar patches in representing the *appearance* of arbitrary objects, they fall short of capturing the *geometrical* configurations of arbitrary objects, especially for objects with articulation. Finding a good representation for objects remains a fundamental problem in computer vision.

6. Scaling up. Larger images, more objects, more complex cues. This might require an efficient non-uniform sampling technique, a compact representation of objects, and most likely a good representation of scene context.

Bibliography

Adams, R. and Bischof, L. (1994). Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6):641–7.

Adelson, E. H. (1999). Lightness perception and lightness illusions. In Gazzaniga, M., editor, *The cognitive neurosciences*, pages 339–51. MIT Press, Cambridge, MA.

Adelson, E. H. and Pentland, A. P. (1996). The perception of shading and reflectance. In Knill, D. and Richards, W., editors, *Perception as Bayesian inference*, pages 409–23. Cambridge University Press, New York.

Alpert, C. J. and Kahng, A. B. (1995a). Multiway partitioning via geometric embeddings, orderings, and dynamic programming. *IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems*, 14(11):1342–58.

Alpert, C. J. and Kahng, A. B. (1995b). Multiway partitioning via geometric embeddings, orderings and dynamic programming. *IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems*, 14(11):1342–58.

Alpert, C. J., Kahng, A. B., and Yao, S.-Z. (1995). Spectral partitioning: the more eigenvectors, the better. In *32nd ACM/IEEE conference on Design automation conference*.

Amir, A. and Lindenbaum, M. (1996). Quantitative analysis of grouping process. In *European Conference on Computer Vision*, pages 371–84.

Amir, A. and Lindenbaum, M. (1998a). A generic grouping algorithm and its quantitative analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(2):168–85.

Amir, A. and Lindenbaum, M. (1998b). Ground from figure discrimination. In *International Conference on Computer Vision*, pages 521–7.

Amir, A. and Lindenbaum, M. (1998c). Grouping-based nonadditive verification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(2):186–92.

Anstreicher, K. and Wolkowicz, H. (2000). On Lagrangian relaxation of quadratic matrix constraints. *SIAM J. Matrix Anal. Appl.*, (1):41–55.

Attneave, F. (1954). Informational aspects of visual perception. *Psychological Review*, 61:183–93.

Barlow, H. (1960). The coding of sensory messages. In *Current problems in animal behaviour*, pages 331–60. Cambridge University Press, Cambridge.

Barnes, E. R. (82). An algorithm for partitioning the nodes of a graph. *SIAM J. Alg. Disc. Meth.*, 3(4):541–50.

Beck, J. (1982). Textural segmentation. In Beck, J., editor, *Organization and representation in perception*, pages 285–317. Hillsdale, NJ: Erlbaum.

Belhumeur, P. (1996). A Bayesian approach to binocular stereopsis. *International Journal of Computer Vision*, 19(3):237–260.

Berg, C., Christensen, J. P. R., and Ressel, P. (1984). *Harmonic analysis on semigroups*. Springer-Verlag.

Bergen, J. R. and Adelson, E. H. (1988). Early vision and texture perception. *Nature*, 333:363–4.

Blake, A. and Isard, M. (1998). *Active contours: the application of techniques from graphics, vision, control theory and statistics to visual tracking of shapes in motion*. Springer-Verlag.

Blake, A. and Zisserman, A. (1987). *Visual reconstruction*. MIT Press, Cambridge, MA.

Borenstein, E. and Ullman, S. (2002). Class-specific, top-down segmentation. In *European Conference on Computer Vision*.

Boykov, Y., Veksler, O., and Zabih, R. (1998). Markov random fields with efficient approximations. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Boykov, Y., Veksler, O., and Zabih, R. (1999). Fast approximate energy minimization via graph cuts. In *International Conference on Computer Vision*.

Bulthoff, I., Bulthoff, H., and Sinha, P. (1998). Top-down influences on stereoscopic depth-perception. *Nature Neuroscience*, 1(3):254–7.

Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–98.

Chan, P. K., Schlag, M. D. F., and Zien, J. Y. (1994). Spectral k-way ratio-cut partitioning and clustering. *IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems*, 13(9):1088–96.

Chung, F. R. K. (1997). *Spectral Graph Theory*. Am. Math. Soc.

Cohen, L. D. (1991). On active contour models and ballons. *Computer Vision, Graphics and Image Processing*, 53(2):211–8.

Deriche, R. (1990). Fast algorithms for low-level vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:78–87.

Dobbins, A. C., Jeo, R. M., Fiser, J., and Allman, J. M. (1998). Distance modulation of neural activity in the visual cortex. *Science*, 281:552–5.

Ferrari, P. A., Frigessi, A., and SA, P. G. D. (1995). Fast approximate maximum a posteriori restoration of multicolour images. *Journal of the Royal Statistics Society, Series B*, 57(3):485–500.

Freeman, W. T. (1996). The generic viewpoint assumption in a Bayesian framework. In Knill, D. C. and Richards, W., editors, *Perception as Bayesian inference*, pages 365–89. Cambridge University Press.

Fukunaga, K. (1990). *Introduction to statistical pattern recognition*. Academic Press.

Gander, W., Golub, G. H., and von Matt, U. (1989). A constrained eigenvalue problem. *Linear Algebra and its applications*, 114/5:815–39.

Gdalyahu, Y., Weinshall, D., and Werman, M. (1998). A randomized algorithm for pairwise clustering. In *Neural Information Processing Systems*, pages 424–30.

Geiger, D. and Kumaran, K. (1996). Visual organization of illusory surfaces. In *European Conference on Computer Vision*, Cambridge, England.

Geiger, D., kuo Pao, H., and Rubin, N. (1998). Salient and multiple illusory surfaces. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–41.

Greig, D. M., Porteous, B. T., and Seheult, A. H. (1989). Exact maximum A Posteriori estimation for binary images. *Journal of the Royal Statistics Society, Series B*, 51(2):271–9.

Grenander, U. and Miller, M. I. (1994). Representations of knowledge in complex systems. *Journal of the Royal Statistics Society, Series B*, 56(4):549–603.

Grossberg, S. and Mingolla, E. (1985). Neural dynamics of form perception: boundary completion, illusory figures, and neon color spreading. *Psychological Review*, 92:173–211.

Hall, K. M. (1970). An r-dimensional quadratic placement algorithm. *Manag. Sci.*, pages 219–29.

Heitger, F. and von der Heydt, R. (1993). A computational model of neural contour processing: Figure-ground segregation and illusory contours. In *International Conference on Computer Vision*, pages 32–40.

Hendrickson, B. and Leland, R. (1995). An improved spectral graph partitioning algorithm for mapping parallel computations. *SIAM Journal on Scientific Computing*, 16(2):452–459.

Hochberg, J. E. and Brooks, V. (1962). Pictorial recognition as an unlearned ability: a study of one child's performance. *American Journal of Psychology*, 75:624–8.

Hoffman, D. D. (1983). The interpretation of visual illusions. 249(6):154–62.

Hong, T. H. and Rosenfeld, A. (1984). Compact region extraction using weighted pixel linking in a pyramid. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(2):222–9.

Illingworth, J. and Kittler, J. (1988). A survey of the Hough transform. *Computer Vision, Graphics and Image Processing*, 44:87–116.

Ishikawa, H. (2003). Exact optimization for markov random fields with convex priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Ishikawa, H. and Geiger, D. (1998). Segmentation by grouping junctions. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Jaakkola, T., Meila, M., and Jebara, T. (1999). Maximum entropy discrimination. In *Neural Information Processing Systems*, volume 12.

Jacobs, D. (1996). Robust and efficient detection of convex groups. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (1):23–37.

Jacobs, D. W. (1992). *Recognizing 3D objects using 2D images*. PhD thesis, Massachusetts Institute of Technology.

Jain, A. (1989). *Fundamentals of digital image processing*. Prentice Hall.

Joachims, T. (1999). Transductive inference for text classification using support vector machines. In *International Conference on Machine Learning*.

Jordan, M. I., editor (1999). *Learning in Graphical Models*. MIT Press.

Julesz, B. (1984). A brief outline of the texton theory of human vision. 7:41–5.

Julesz, B. (1986). Texton gradients: the texton theory revisited. *Biological Cybernetics*, 54:245–51.

Kamvar, S. D., Klein, D., and Manning, C. D. (2002). Interpreting and extending classical agglomerative clustering algorithms using a model-based approach. In *International Conference on Machine Learning*.

Kanizsa, G. (1979). *Organization in vision*. Praeger Publishers.

Kass, M., Witkin, A., and Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, pages 321–331.

Kelly, F. and Grossberg, S. (2000). Neural dynamics of 3-D surface percpetion: figure-ground separation and lightness perception. *Perception and Psychophysics*.

Knierim, J. J. and van Essen, D. (1992). Neuronal responses to static texture patterns in area V1 of the alert macaque monkey. *Journal of Neurophysiology*, 67(4):961–80.

Knill, D. C. and Richards, W., editors (1996). *Perception as Bayesian inference*. Cambridge University Press.

Koffka, K. (1935). *Principles of Gestalt Psychology*. A Harbinger Book, Harcourt Brace & World Inc.

Kolmogorov, V. and Zabih, R. (2002a). Multi-camera scene reconstruction via graph cuts. In *European Conference on Computer Vision*.

Kolmogorov, V. and Zabih, R. (2002b). What energy functions can be minimized via graph cuts? In *European Conference on Computer Vision*.

Lamme, V. (1995). The neurophysiology of figure-ground segregation in primary visual cortex. *Jounral of neuroscience*, 10:649–69.

Lee, T. S. and Mumford, D. (2003). Hierarchical bayesian inference in the visual cortex. *Journal of Optical Society of America*.

Lee, T. S., Mumford, D., Romero, R., and Lamme, V. (1998). The role of primary visual cortex in higher level vision. *Vision Research*, 38:2429–54.

Li, Z. (2000). Pre-attentive segmentation in the primary visual cortex. *Spatial Vision*, 13(1):25–50.

Lowe, D. (1984). *Perceptual organization and visual recognition*. PhD thesis, Stanford University, Department of Computer Science.

Madarasmi, S., Pong, T.-C., and Kersten, D. (1994). Illusory contour detection using MRF models. In *IEEE International Conference on Neural Networks*, volume 7, pages 4343–8.

Mahamud, S. (2002). *Discriminative distance measures for object detection*. PhD thesis, Carnegie Mellon University.

Mahamud, S., Williams, L. R., Thornber, K. K., and Xu, K. (2003). Segmentation of multiple salient closed contours from real images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(4).

Malik, J., Belongie, S., Leung, T., and Shi, J. (2001). Contour and texture analysis for image segmentation. *International Journal of Computer Vision*.

Malik, J. and Perona, P. (1990). Preattentive texture discrimination with early vision mechanisms. *Journal of Optical Society of America*, 7:923–32.

Marr, D. (1982). *Vision*. CA: Freeman.

Martin, D., Fowlkes, C., and Malik, J. (2002). Learning to detect natural image boundaries using brightness and texture. In *Neural Information Processing Systems*.

Martin, D., Fowlkes, C., Tal, D., and Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *International Conference on Computer Vision*.

McInerney, T. and Terzopoulos, D. (1996). Deformable models in medical image analysis: a survey. *Medical Image Analysis*, 1(2):91–108.

Meila, M. and Shi, J. (2001). Learning segmentation with random walk. In *Neural Information Processing Systems*.

Mumford, D. (1993). Elastica and computer vision. In Bajaj, C. L., editor, *Algebraic geometry and its applications*. Springer-Verlag.

Mumford, D. (1996). Pattern theory: a unifying perspective. In Knill, D. C. and Richards, W., editors, *Perception as Bayesian inference*, pages 25–61. Cambridge University Press.

Mumford, D. and Shah, J. (1985). Boundary detection by minimizing functionals. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 22–6.

Nakayama, K. and Shimojo, S. (1992). Experiencing and perceiving visual surfaces. *Science*, 257:1357–63.

Nakayama, K., Shimojo, S., and Silverman, G. H. (1989). Stereoscopic depth: its relation to image segmentation, grouping, and the recognition of occluded objects. *Perception*, 18:55–68.

Ng, A. Y., Jordan, M., and Weiss, Y. (2002). On spectral clustering: Analysis and an algorithm. In *Neural Information Processing Systems*.

Nigam, K., McCallum, A. K., Thrun, S., and Mitchell, T. (1999). Text classification from labeled and unlabeled documents using EM. *Machine Learning*, pages 1–34.

Nitzberg, M., Mumford, D., and Shiota, T. (1993). *Filtering, segmentation and depth*. Springer-Verlag.

Nothdurft, H.-C. (1993). The role of features in preattentive vision: comparison of orientation, motion and color cues. *Vision Research*, 33(14):1937–58.

Nothdurft, H.-C. (1997). Different approaches to the coding of visual segmentation. In Harris, L. and Kjenkius, M., editors, *Computational and psychophysical mechanisms of visual coding*. Cambridge University Press, New York.

Palmer, S. E. (1999). *Vision science: from photons to phenomenology*. MIT Press.

Perona, P. and Freeman, W. (1998). A factorization approach to grouping. In *European Conference on Computer Vision*, pages 655–70.

Peterson, M. A. (1994). Object recognition processes can and do operate before figure-ground organization. *Current Directions in Psychological Science*, 3:105–11.

Puzicha, J., Hofmann, T., and Buhmann, J. (1998). Unsupervised texture segmentation in a deterministic annealing framework. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):803–18.

Rimey, R. D. (1993). Control of selective perception using Bayes nets and decision theory. Technical Report TR468.

Ronfard, R. (1994). Region-based strategies for active contour models. *International Journal of Computer Vision*, 13(2).

Roweis, S. and Saul, L. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–6.

Roy, S. and Cox, I. J. (1998). A maximum-flow formulation of the n-camera stereo correspondence problem. In *International Conference on Computer Vision*.

Rumelhart, D. E. and McClelland, J. L. (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*. MIT Press.

Sagi, D. and Julesz, B. (1987). Short-range limitations on detection of feature differences. *Spatial Vision*, 2:39–49.

Sahoo, P. K., Soltani, S., and Wong, A. K. C. (1988). A survey of thresholding techniques. *Computer Vision, Graphics and Image Processing*, 41:233–60.

Schneiderman, H. and Kanade, T. (2002). Object detection using the stattistics of parts. *International Journal of Computer Vision*.

Scott, G. L. and Longuet-Higgins, H. C. (1991). Feature grouping by relocalization of eigenvectors of the proximity matrix. In *Proceedings of the British Machine Vision Conference*, pages 103–8.

Sethian, J. (1996). *Level set methods*. Cambridge University Press.

Sharon, E., Brandt, A., and Basri, R. (2000). Fast multiscale image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 70–7.

Shi, J. (1998). *Perceptual organization and image segmentation*. PhD thesis, Department of Computer Science, University of California at Berkeley.

Shi, J. and Malik, J. (1997). Normalized cuts and image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 731–7.

Shi, J. and Malik, J. (1998). Self inducing relational distance and its application to image segmentation. In *European Conference on Computer Vision*, pages 528–43.

Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905.

Sidenbladh, H. and Black, M. (2001). Learning image statistics for Bayesian tracking. In *International Conference on Computer Vision*.

Szummer, M. and Jaakkola, T. (2001). Partially labeled classification with Markov random walks. In *Neural Information Processing Systems*, volume 14.

Treisman, A. (1985). Preattentive processing in vision. *Computer Vision, Graphics and Image Processing*, 31:156–77.

Tu, Z. and Zhu, S. C. (2002). Image segmentation by data-driven Markov chain Monte Carlo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:657–73.

Ullman, S. (1976). Filling-in the gaps: the shape of subjective contours and a model for their generation. *Biological Cybernetics*, 25:1–6.

van Essen, D. C. (1985). Functional organization of primate visual cortex. In *Cerebral Cortex*, pages 259–329.

Vecera, S. P. and O'Reilly, R. C. (1998). Figure-ground organization and object recognition processes: An interactive account. *Journal of Experimental Psychology: Human Perception and Performance*, pages 441–62.

Wagstaff, K., Cardie, C., Rogers, S., and Schroedl, S. (2000). Clustering with instance-level constraints. In *International Conference on Machine Learning*.

Wagstaff, K., Cardie, C., Rogers, S., and Schroedl, S. (2001). Constrained K-means clustering with background knowledge. In *International Conference on Machine Learning*.

Weiss, Y. (1999). Segmentation using eigenvectors: a unifying view. In *International Conference on Computer Vision*, pages 975–82.

Wertheimer, M. (1938). Laws of organization in perceptual forms (partial translation). In Ellis, W. B., editor, *A sourcebook of Gestalt Psychology*, pages 71–88. Harcourt Brace and company.

Wildes, R. (1991). Direct recovery of three-dimensional scene geometry from binocular stereo disparity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 761–74.

Williams, L. R. and Jacobs, D. W. (1997). Local parallel computation of stochastic completion fields. *Neural computation*, 9(4):859–81.

Witkin, A. and Tenenbaum, J. M. (1983). On the role of structure in vision. In Beck, Hope, and Rosenfeld, editors, *Human and machine vision*, pages 481–543. Academic Press, New York.

Witkin, A. P. (1983). Scale-space filtering. In *Proc.* 8*th International Joint Conference on AI*, pages 1019–22.

Wolfson, H. J. and Rigoutsos, I. (1997). Geometric hashing: an overview. *IEEE Computational Science and Engineering*, 4(4):10–21.

Wu, Z. and Leahy, R. (1993). An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:1101–13.

Xu, C., Pham, D. L., and Prince, J. L. (2000). *Medical image segmentation using deformable models*, pages 129–74. SPIE.

Yarbus, A. L. (1967). *Eye movements*. Plenum, New York.

Yu, S. X., Gross, R., and Shi, J. (2002). Concurrent object recognition and segmentation by graph partitioning. In *Neural Information Processing Systems*.

Yu, S. X., Lee, T. S., and Kanade, T. (2001). A hierarchical Markov random field model for figure-ground segregation. In Figueiredo, M., Zerubia, J., and Jain, A. K., editors, *Lecture Notes in Computer Science*, number 2134, pages 118–33. Springer-Verlag.

Yu, S. X. and Shi, J. (2001a). Grouping with bias. In *Neural Information Processing Systems*.

Yu, S. X. and Shi, J. (2001b). Perceiving shapes through region and boundary interaction. Technical Report CMU-RI-TR-01-21, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA.

Yu, S. X. and Shi, J. (2001c). Segmentation with pairwise attraction and repulsion. In *International Conference on Computer Vision*.

Yu, S. X. and Shi, J. (2003). Multiclass spectral clustering. *submitted*.

Yuille, A. L., Cohen, D. S., and Hallinan, P. W. (1989). Feature extraction from faces using deformable templates. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 104–9.

Zhou, H., Friedman, H., and von der Heydt, R. (2000). Coding of border ownership in monkey visual cortex. *Journal of Neuroscience*, 20(17):6594–611.

Zhu, S. C. (1999). Embedding Gestalt laws in Markov random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11).

Zhu, S. C., Wu, Y. N., and Mumford, D. (1998). Filters, random field and maximum entropy: — towards a unified theory for texture modeling. *International Journal of Computer Vision*, 27(2):1–20.

Zhu, S. C. and Yuille, A. (1996). Region competition: unifying snakes, region growing, and Bayes/MDL for multi-band image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9):884–900.