Multi-Class Constrained Normalized Cut With Hard, Soft, Unary and Pairwise Priors and Its Applications to Object Segmentation

Han Hu, Jianjiang Feng, *Member, IEEE*, Chuan Yu, and Jie Zhou, *Senior Member, IEEE*

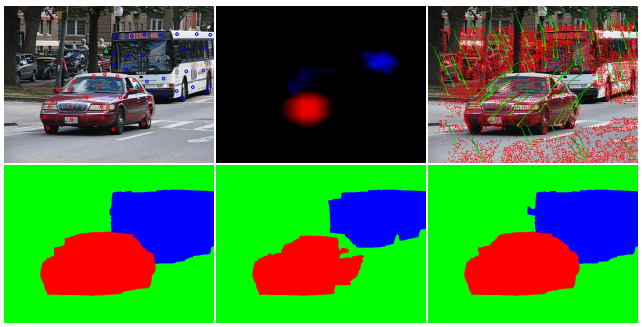
***Abstract*— Normalized cut is a powerful method for image segmentation as well as data clustering. However, it does not perform well in challenging segmentation problems, such as segmenting objects in a complex background. Researchers have attempted to incorporate priors or constraints to handle such cases. Available priors in image segmentation problems may be hard or soft, unary or pairwise, but only hard must-link constraints and two-class settings are well studied. The main difficulties may lie in the following aspects: 1) the nontransitive nature of cannot-link constraints makes it hard to use such constraints in multi-class settings and 2) in multi-class or pairwise settings, the output labels have inconsistent representations with given priors, making soft priors difficult to use. In this paper, we propose novel algorithms, which can handle both hard and soft, both unary and pairwise priors in multi-class settings and provide closed form and efficient solutions. We also apply the proposed algorithms to the problem of object segmentation, producing good results by further introducing a spatial regularity term. Experiments show that the proposed algorithms outperform the state-of-the-art algorithms significantly in clustering accuracy. Other merits of the proposed algorithms are also demonstrated.**

***Index Terms*— Constrained spectral clustering, object segmentation, unary priors, pairwise priors.**

I. Introduction

IMAGE segmentation is an essential task in computer vision. Typically, it works in a bottom-up manner by detecting contours or merging pixels using brightness, color or texture cues. However, bottom-up image segmentation algorithms often cannot provide meaningful results because of the large semantic gap between concept-level objects and low-level feature cues. For example, in Fig. 1, it is hard to know that the window, red body and white licence plate belong to a same object, car, by exploiting only low-level cues. As a result, the best thing that a low-level image segmentation algorithm can do is to output hierarchical segments (see Fig. 1 for the results of a state-of-the-art low-level image segmentation algorithm [1]).

Fig. 1. An example of low level image segmentation. Left: an example image; middle and right: bottom up segmentation results (represented by average color of each segment) at two scales by the method in [1].



(a) (b) (c)

Fig. 2. Three types of priors used in this study to improve segmentation performance. (a) Hard unary priors (colored circle points) from user interaction (top row) and our segmentation result (bottom row); (b) soft unary priors from object detectors (*car* (red) and *bus* (blue)) [12] (top row), and our segmentation result (bottom row); (c) soft pairwise priors obtained by analyzing the point trajectories from an image sequence[1] (top row), and our segmentation result (bottom row).

To achieve meaningful image segmentation results, which is the objective of object segmentation algorithms [2]–[4], it is necessary to incorporate high-level knowledge into the segmentation framework. Three types of priors in object segmentation are illustrated in Fig. 2. Early researches usually utilize exact unary priors, which are input by human and indicate whether a pixel belongs to the same object or not, to guide segmentation, also referred as interactive image segmentation [2], [5], [6] (see Fig. 2(a)).

One can also utilize continuous or noisy unary priors. Such priors may come from an object detector and are typically in the form of a response image which indicates the possibilities of an object existence [3] (see Fig. 2(b)). Different from interactive segmentation, no manual input is needed here.

Manuscript received June 15, 2012; revised November 29, 2012; accepted June 16, 2013. Date of publication July 3, 2013; date of current version September 12, 2013. This work was supported by the National Natural Science Foundation of China under Grants 61225008, 61020106004, 61005023, and 61021063, and the Ministry of Education of China under Grant 20120002110033. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Pascal Frossard.

The authors are with the State Key Laboratory on Intelligent Technology and Systems, Tsinghua National Laboratory for Information Science and Technology, Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: huh04@mails.thu.edu.cn; jfeng@tsinghua.edu.cn; chuanyu08@mails.thu.edu.cn; jzhou@tsinghua.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2013.2271865

[1] To make the figure clear, we only draw trajectories of a few points in green color.

Another type of prior useful for image segmentation is pairwise constraint which indicates whether two pixels belong to the same segment or not, referred to as must-link and cannot-link constraints respectively. These constraints could be hard ones input by user interaction or soft ones contributed by other more reliable features. Consider Fig. 2(c). When an image sequence is available, we can obtain a set of point trajectories using optical flow [7]. Psychophysical studies have shown that such motion information is moderately reliable for segmenting objects and plays a key role in helping infants and people newly recovering from congenitally blind to get mature vision such as segmenting and parsing the scenes [8]. Although there have been a lot of studies for trajectory-based segmentation [4], [9], [10], their results are still not sufficiently robust or accurate to be directly regarded as unary ground truth seeds. Incorporating motion cues into affinity matrix calculation [11] is also inappropriate since trajectories can not be accurately extracted for many pixels especially in smooth areas. Nevertheless, we could extract a sparse set of trajectories with sufficient accuracy [7]; measure the probability of two trajectories belonging to a same segment by comparing their motion patterns; and use them as soft pairwise priors for image segmentation.

In this paper, we attempt to incorporate all the above priors into the normalized cut framework [11], [13], which is one of the most important and popular methods for data clustering and image segmentation. Compared with traditional clustering algorithms such as k-means and hierarchical clustering, normalized cut has mainly the following advantages: deterministic polynomial-time solution, the ability to model arbitrary shaped clusters, and its equivalence to certain graph partitioning, spectral embedding and random walk problems [13]. Inspired by the success of normalized cut, how to incorporate priors or constraints into normalized cut has attracted more and more attention. Based on the way they encode priors, the existing studies can be roughly grouped into two categories. The first category [14]–[16] directly manipulates the affinity matrix according to given constraints, and then applies the unconstrained normalized cut algorithm on the data points. The second category [2], [17]–[21] models the priors as constraints or penalties into the unconstrained normalized cut problem, and achieves clustering by solving the new optimization problem. Since priors are directly linked to the clustering results, the second category usually performs better than the first one. However, none of the above methods can handle all types of the mentioned priors well in multi-class settings because of the following reasons.

Firstly, it is difficult to encode between-cluster or cannot-link constraints into the multi-class normalized cut problem due to their nontransitive property [2], [19]. As a result, cannot-link constraints are usually either discarded [2] or used in only two-class problem [20], [21]. Although a few algorithms have been proposed to encode hard cannot-link constraints in multi-class settings [14], [15], [19], [22], they are still not well exploited.

Secondly, it is difficult to handle soft priors in multi-class or pairwise settings due to inconsistent representations of priors and output labels. Most of the mentioned approaches [2], [14], [15], [19] can handle only hard priors. Although the method of [21] successfully encodes soft priors, it can be applied in only two-class (e.g., foreground segmentation of an image) normalized cut problem with unary priors.

We propose novel algorithms, Multi-class Constrained Normalized Cut with Unary Priors (MCNC-UP) and Multi-class Constrained Normalized Cut with Pairwise Priors (MCNC-PP), in order to settle the above difficulties in utilizing unary and pairwise priors in multi-class clustering problems respectively. Two key points of the proposed algorithms are: 1) priors are encoded by a correlation function, which makes soft and cannot-link priors tractable and also takes merits of penalizing all points instead of isolated ones; 2) the matrix-form pairwise priors are decomposed into columns, where each column can be regarded as a unary prior vector for a 2-class constrained normalized cut problem. In this way, the matrix-form priors and vector-form labels are made consistent with each other such that they can be modeled by correlation functions.

The proposed algorithms also have the merit of closed form solutions. An interesting property of such solution is that if the priors change, the optimal solution can be computed efficiently without performing another eigen-decomposition, which is usually computationally expensive for large-scale data. This property is extremely useful for interactive image segmentation.

In addition, we apply the proposed MCNC algorithms to the problem of object segmentation by exploiting 3 types of priors. We find that when combined with a Markov Random Field (MRF) framework, the MCNC algorithms significantly outperform related algorithms on challenging image datasets.

Extensive experiments on synthetic data, public clustering datasets and image segmentation datasets demonstrate the advantages of the proposed algorithms. The codes are publicly available at https://sites.google.com/site/hanhushomepage/projects-researches.

The remainder of this paper is organized as follows: in Section II, we review the normalized cut problem and its spectral relaxations; in Section III and IV, the MCNC-UP and MCNC-PP algorithms are presented in detail; after that, we propose a MRF framework to apply the proposed MCNC algorithms to the problem of object segmentation by using 3 types of priors; extensive experiments on synthetic data, public clustering datasets and image segmentation datasets are conducted in Section VI; finally, conclusions are made in Section VII.

II. Normalized Cut

Normalized cut approach regards an image as a weighted graph $G(V, E, W)$, with the vertex set V representing all n pixels in the image and the edge set E together with the edge weights $W : V \times V \rightarrow \mathbb{R}_{+}^{n \times n}$ representing similarities between pixels. The design of W is very critical for improving the performance of image segmentation. The seminar work of Shi and Malik [11] uses simple feature cues to construct W. A more sophisticated measure in [1] combines several low-level cues to detect contours and constructs the weight matrix using the intervening contour cue.

The task of image segmentation is to partition the pixel set V into c segments $\{C_i\}_{i=1}^{c}$, with $|C_i| = n_i$. Define the cut as, $\text{cut}(C_1, C_2) = \sum_{i\in C_1, j\in C_2} W_{ij}$, and the volume as $\text{vol}(C) = \sum_{i\in C} \mathbf{d}_i$, with degree $\mathbf{d}_i = \sum_{j\in V} W_{ij}$. Then normalized cut achieves segmentation by minimizing the total cut balanced with the cluster volume [11], [13],

$$J_{\text{ncut}} = \sum_{i=1}^{c} \frac{\text{cut}(C_i, \overline{C_i})}{\text{vol}(C_i)}, \tag{1}$$

where $\overline{C_i} = V \setminus C_i$. Since it is NP-hard to exactly optimize the above normalized cut problem [11], most approaches compute an approximate solution by spectral relaxation. The relaxations are slightly different for two-class segmentation ($c = 2$) and mutli-class segmentation ($c > 2$) [13], and we discuss them respectively.

Two-Class Normalized Cut: We denote the indicator vector $\mathbf{y}$ as: $\mathbf{y}(i) = \sqrt{\frac{\text{vol}(C_2)}{\text{vol}(C_1)}}$, if $V_i \in C_1$; $\mathbf{y}(i) = -\sqrt{\frac{\text{vol}(C_1)}{\text{vol}(C_2)}}$, if $V_i \in C_2$. Denote the degree matrix as $D = \text{diag}(\mathbf{d})$, and Laplacian matrix as $L = D - W$. One can check that $(D\mathbf{y})^T\mathbf{1} = 0$, $\mathbf{y}^T D\mathbf{y} = \text{vol}(V)$, and $\mathbf{y}^T L\mathbf{y} = \text{vol}(V) J_{\text{ncut}}$. Thus the two-class normalized cut optimization problem can be relaxed as

$$\min_{\mathbf{y}^T D\mathbf{y}=\text{vol}(V), D\mathbf{y}\perp\mathbf{1}} \mathbf{y}^T L\mathbf{y}. \tag{2}$$

We substitute $\mathbf{x} = D^{1/2}\mathbf{y}$ and denote normalized Laplacian matrix $L_{\text{sym}} = D^{-1/2} L D^{-1/2}$ with eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ and corresponding eigenvectors $\mathbf{u}_1, \ldots, \mathbf{u}_n$. Then eq. (2) becomes,

$$\min_{\mathbf{x}^T\mathbf{x}=\text{vol}(V), \mathbf{x}\perp D^{1/2}\mathbf{1}} \mathbf{x}^T L_{\text{sym}}\mathbf{x}. \tag{3}$$

If V is connected, then the optimal $\mathbf{x}$ is given by $\mathbf{u}_2$, the second eigenvector of L_{sym}, and the optimal $\mathbf{y}$ is given by $D^{-1/2}\mathbf{u}_2$.

Multi-Class Normalized Cut: Denote indicator matrix $Y \in \mathbb{R}^{n\times c}$ as: $Y_{ij} = \frac{1}{\sqrt{\text{vol}(C_j)}}$, if $V_i \in C_j$; $Y_{ij} = 0$, if $V_i \in \overline{C_j}$. Substitute $X = D^{1/2}Y$, and then similar to two-class normalized cut, the multi-class normalized cut optimization can be relaxed as

$$\min_{X^T X = I} \text{tr}(X^T L_{\text{sym}} X). \tag{4}$$

Denote $U = [\mathbf{u}_1, \ldots, \mathbf{u}_c]$. Then the optimal X^* shares a same Stiefel manifold with U as $X^* = UR$, where $R \in \mathbb{R}^{c\times c}$ is an orthogonal matrix. The final segmentation could be obtained by either k-means or spectral rotation.

III. Multi-Class Constrained Normalized Cut With Unary Priors

We encode the unary priors $S \in \mathbb{R}^{n\times c}$ by correlation constraints, $\text{diag}((X^T S)) \geq \kappa \text{diag}(I)$. To make the correlation functions meaningful, S should be consistent with the output indicator matrix X in representation. To satisfy $S^T S = I$, S is set as

$$S_{ij} = \begin{cases} \sqrt{\frac{\mathbf{d}_i}{\text{vol}(C_j^p)}}, & \text{if } V_i \in C_j^p \\ 0, & \text{otherwise,} \end{cases} \tag{5}$$

where C_j^p denotes the set of priors belonging to the j^{th} segment. Adding the correlation constraints into the optimization framework of eq. (4), we get

$$\begin{aligned} &\min_X \text{tr}(X^T L_{\text{sym}} X) \\ &s.t.\ \ X^T X = I, \text{diag}(X^T S) \geq \kappa \text{diag}(I). \end{aligned} \tag{6}$$

However, due to non-convexity of the constraints, it is very hard to solve the above problem. To make eq. (6) tractable, we replace $X^T X = I$ by two constraints: $\text{diag}(X^T X) = \text{diag}(I)$ and $X^T D^{1/2}\mathbf{1} = \mathbf{0}$. We get

$$\begin{aligned} &\min_X \text{tr}(X^T L_{\text{sym}} X) \\ &s.t.\ \ \text{diag}(X^T X) = \text{diag}(I), X^T D^{1/2}\mathbf{1} = \mathbf{0}, \\ &\qquad \text{diag}(X^T S) \geq \kappa \text{diag}(I). \end{aligned} \tag{7}$$

Accordingly, the priors are adjusted to satisfy the two new constraints:

$$S_{ij} = \begin{cases} \sqrt{\frac{\mathbf{d}_i \text{vol}(\overline{C_j^p})}{\text{vol}(C_j^p)\text{vol}(C^p)}}, & \text{if } V_i \in C_j^p \\ -\sqrt{\frac{\mathbf{d}_i \text{vol}(C_j^p)}{\text{vol}(\overline{C_j^p})\text{vol}(C^p)}}, & \text{if } V_i \in \overline{C_j^p} \\ 0, & \text{otherwise,} \end{cases} \tag{8}$$

where C^p denotes the set of priors and $\overline{C_j^p} = C^p \setminus C_j^p$ denotes the pixels in C^p but not belonging to the j^{th} segment.

Note that although the orthogonal constraints are discarded in the new optimization, the columns of X are still complementary thanks to the complementary columns of priors S, the correlation constraints, and the orthogonality of X to the trivial vector $D^{1/2}\mathbf{1}$.

We have the following theorem.

Theorem 1: Let G be a connected graph and let $0 \leq \kappa \leq 1$ be a correlation parameter. Then, there is an optimal solution, X^*, to the problem of eq. (7) such that $X^* = (L_{\text{sym}} - \alpha L_{\text{n}})^{\dagger} SB$, where $\alpha \in (-\infty, \lambda_2(G))$ is a function that depends on κ; B is a diagonal matrix to ensure $\text{diag}(X^T X) = \text{diag}(I)$ being satisfied; $L_{\text{n}} = I - D^{1/2}\mathbf{1}\mathbf{1}^T D^{1/2}$; and † is the pseudo-inverse operator.

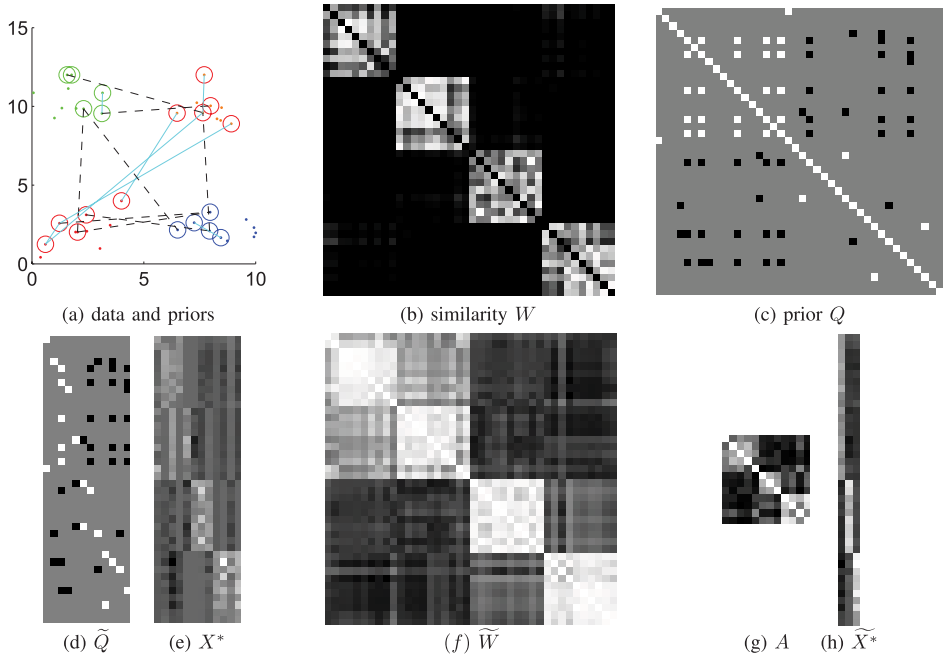
Proof. We include the proof in Appendix. □

MCNC-UP Algorithm:

Since α is a function depending on κ, we can use α instead of κ as the problem-dependent parameter. In this case, Theorem 1 provides a closed form solution for eq. (7), and if the spectrum of L_{sym} has already been computed, one can compute the optimal solution in very little additional effort as: $X^* = \sum_{i=2}^{n} \frac{1}{\lambda_i - \alpha} \mathbf{u}_i \mathbf{u}_i^T SB$.

One can find that each column of X^* is a weighted combination of the eigenvectors, and the weight of each eigenvector is proportional to the correlation with the corresponding seed column and inversely proportional to $\lambda_i - \alpha$. This indicates that the eigenvectors which are well correlated with the priors are up weighted. Generally, $\lambda_i - \alpha$ grows quickly, and one can get a good approximation by only reserving the first K eigenvectors.

To the present, we have obtained X^*, which is a $n \times c$ matrix. To generate the final labels, one can directly select the index of the maximum value in each row as its final label. The MCNC-UP algorithm is summarized in Algorithm 1.



(a) data and priors (b) similarity W (c) prior Q

(d) $\widetilde{Q}$ (e) X^* (f) $\widetilde{W}$ (g) A (h) $\widetilde{X^*}$

Fig. 3. A toy example for illustrating MCNC-PP algorithm. There are four groups of data points (colored dots in (a), with similarity matrix W shown in (b)), and we suppose the red and orange points are from the same cluster. We generate several must-link (cyan solid lines in (a) with two circles on the two ends) and cannot-link (black dashed lines in (a)) constraints (see (c) for the prior matrix Q and (d) for merged priors $\widetilde{Q}$). Each column of $\widetilde{Q}$ can be viewed as the unary constraints for a two-class clustering problem, and the corresponding optimal solution, X^*, is shown as (e). Two schemes are proposed to obtain the final label vector from X^*: in (f), a similarity graph for data points is constructed from X^*, which is used to generate point clusters; in (g-h), a similarity graph for priors is constructed (see (g)) and the priors are clustered into c clusters accordingly, whose centers (see (h)) are used to yield point labels.

Algorithm 1 Multi-Class Constrained Normalized Cut With Unary Priors (MCNC-UP)

Require: Graph $G(V, E, W)$, unary priors S, and a correlation parameter $\alpha \in (-\infty, \lambda_2(G))$

1: Compute the eigenvectors $U = [\mathbf{u}_1, \ldots, \mathbf{u}_K]$ of the normalized Laplacian matrix L_{sym} corresponding to the K smallest eigenvalues $\lambda_1, \ldots \lambda_K$, with $L_{\text{sym}} = D^{-1/2}(D - W)D^{-1/2}$ and $D(i,i) = \sum_{j \in V} W_{ij}$,

2: compute $X^* = \sum_{i=2}^{K} \frac{1}{\lambda_i - \alpha} \mathbf{u}_i \mathbf{u}_i^T SB$, where diagonal matrix $B > 0$ is to ensure $\text{diag}(X^T X) = \text{diag}(I)$,

3: **return** label $\mathbf{l}$, with $\mathbf{l}_j$ being the index of the maximum value in the j^{th} row of X^*.

IV. Constrained Normalized Cut With Pairwise Priors

Pairwise prior, which specifies whether two nodes belong to the same segment or not, is a weaker but more general constraint than the unary one. Pairwise priors could be encoded by an $n \times n$ matrix Q, where $Q_{ij} = +1$ if pixels i and j are from the same segment, namely must-link constraint; $Q_{ij} = -1$ if pixels i and j are from different segments, namely cannot-link constraint; $Q_{ij} = 0$ if no prior is available. In the case of soft priors, we have $|Q_{ij}| \leq 1$ with $|Q_{ij}|$ denoting the confidence score. We also set $Q(i,i) = +1$ since a node definitely belongs to the same segment with itself.

It is obvious that unary priors could lead to pairwise ones. But the reverse process is usually infeasible and thus Algorithm 1 is helpless in the case of pairwise priors. In the following, we will develop a novel algorithm to encode pairwise priors into the normalized cut framework.

We first have a look at a toy example shown in Fig. 3. By checking the prior matrix Q column by column, we could observe that the j^{th} column of Q, $\mathbf{q}_j$, actually provides a set of two-class unary constraints, where the first class contains the points which share the same class with point j and the second class contains the points that do not belong to the same class as point j. Thus we get

$$\begin{aligned} &\min_{\mathbf{x}} \ \text{tr}(\mathbf{x}^T L_{\text{sym}} \mathbf{x}) \\ &s.t. \ \ \|x\| = 1, \mathbf{x}^T D^{1/2} \mathbf{1} = 0, \mathbf{x}^T \mathbf{q}_j \geq \kappa. \end{aligned} \tag{9}$$

For each column which contains pairwise constraints, we can construct an independent optimization problem as eq. (9). By merging all of these optimizations, we get

$$\begin{aligned} &\min_{X} \ \text{tr}(X^T L_{\text{sym}} X) \\ &s.t. \ \ \text{diag}(X^T X) = \text{diag}(N), X^T D^{1/2} \mathbf{1} = \mathbf{0}, \\ &\qquad \text{diag}(X^T \widetilde{Q}) \geq \kappa \text{diag}(N), \end{aligned} \tag{10}$$

where each column $\widetilde{\mathbf{q}}_j$ of $\widetilde{Q} \in \mathbb{R}^{n \times m}$ encodes the priors contained in the j^{th} transitive closure component [23] constructed from Q according to the transitivity of must-link constraints; N is a cardinality matrix with $N(j,j) = |\widetilde{\mathbf{q}}_j|$, which gives the columns with more priors higher weights in the final decision.

Similar as Theorem 1, eq. (10) has an optimal solution as $X^* = (L_{\text{sym}} - \alpha L_{\text{n}})^{\dagger} \widetilde{Q} B$, where $\alpha \in (-\infty, \lambda_2(G))$ is a function that depends on κ; $B > 0$ is a diagonal matrix to ensure that $\text{diag}(X^{*T} X^*) = \text{diag}(N)$ is satisfied; and L_{n} and † have the same meaning as in Theorem 1. As a combination of the original graph and the priors, X^* provides rich information

Algorithm 2 Multi-Class Constrained Normalized Cut With Pairwise Priors (MCNC-PP)

Require: Graph $G(V, E, W)$, pairwise priors Q and a correlation parameter $\alpha \in (-\infty, \lambda_2(G))$

1: Compute the eigenvectors $U = [\mathbf{u}_1, \ldots, \mathbf{u}_K]$ of the normalized Laplacian matrix L_{sym} corresponding to the K smallest eigenvalues $\lambda_1, \ldots, \lambda_K$,
2: if the prior Q is hard, compute the transitive closure components from Q to construct a merged matrix $\widetilde{Q}$; otherwise omit this step,
3: compute $X^* = \sum_{i=2}^{K} \frac{1}{\lambda_i - \alpha} \mathbf{u}_i \mathbf{u}_i^T \widetilde{Q} B$,
4: use either of the following two schemes to get output labels:
5: **scheme (a)**: construct a new graph $\widetilde{G}$ for V as $\widetilde{W_{ij}} = (1 + X_i^* X_j^{*T} / \|X_i^*\| \|X_j^*\|)/2$, where X_i^* is the i^{th} row of X^*, and run normalized cut algorithm to get output labels $\mathbf{l}$,
6: **scheme (b)**: compute similarities between priors: $A_{ij} = (1 + \mathbf{x}_i^{*T} \mathbf{x}_j^* / \|\mathbf{x}_i^*\| \|\mathbf{x}_j^*\|)/2$, where $\mathbf{x}_i^*$ is the i^{th} column of X^*, and cluster the priors into c clusters. The output labels $\mathbf{l}$ are set as the indexes of the maximum values for the rows of cluster center matrix $\widetilde{X^*} \in \mathbb{R}^{n \times c}$.

to generate the output label vector. We present two different schemes as listed in Algorithm 2. In scheme (a), a new graph is constructed from X^* which directly segments the points. In scheme (b), X^* helps clustering prior columns, with each cluster containing priors (considering the positive elements) from one of the segments, and the averaged X^* for each cluster can be regarded as a soft indicator vector for the cluster. The time complexity of scheme (a) is $\mathcal{O}(n^2 m + n^2 K)$, with $\mathcal{O}(n^2 m)$ for constructing the new graph and $\mathcal{O}(n^2 K)$ for normalized cut. Similarly, the time complexity of scheme (b) is $\mathcal{O}(m^2 n + m^2 K)$. Since in most cases, sample number n is much larger than prior column number m, scheme (b) is much faster than (a), and for this reason, we use scheme (b) for motion guided image segmentation.

V. Object Segmentation

A. Three Types of Priors

We apply the proposed algorithms to the problem of object segmentation by utilizing three different types of priors, which are from interactive labeling (hard unary priors), object detection (soft unary priors) and motion analysis (soft pairwise priors), respectively.

Interaction Priors: Generally, only foreground objects need labeling since the background can be determined by the complement of foregrounds. This works well for most areas, but fails when foregrounds have similar colors or textures with background. For such challenging areas, we add a few background labels to correct the errors (see Fig. 4).

Object Detection Priors: We first apply pre-specified object detectors on the image using the method in [12]. The output of each detector is response images of multiple resolutions. We resize all images to the resolution of the original image, and obtain a single image by taking the maximum value at each pixel. The combined response image of each detector constitutes one column of the soft priors.

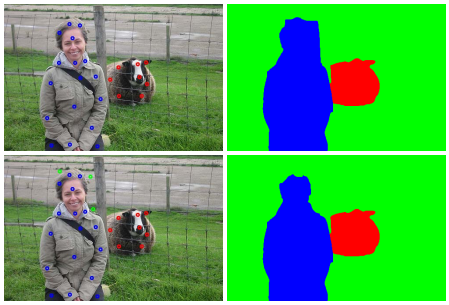


Fig. 4. Interactive segmentation with different labeled points. Top row: interactive segmentation result with labeled points from foregrounds only; bottom row: the ambiguity between the hair and backgrounds is cleared by adding a few labeled background points.

Fig. 5. Trajectory based segmentation results on *cars10* video in Hopkins155 datasets with 3, 10 and 20 segments respectively. This example indicates that trajectory grouping result cannot serve as hard prior for object segmentation.

Pairwise Motion Priors: We compute point trajectories using the approach in [7]. Since the trajectories are very dense ($\sim 10^4$) and short (6 frames are used), we could approximate the motions by translational models and measure pairwise priors based on motion differences:

$$Q_{ij} = \exp\left(-\frac{2\|\mathbf{u}_i - \mathbf{u}_j\|_2^2}{\sigma_i \sigma_j} \right), \tag{11}$$

where $\mathbf{u}_i$ is the motion vector of the i^{th} trajectory; $\|\cdot\|_2$ is the Euclidean distance operator; and σ_i is the standard deviation of all distances associated to the i^{th} trajectory. Q can be used as a similarity matrix to directly cluster trajectories. However, it often fails due to noise or tiny motion differences. For example, considering *cars10* video in Hopkins155 datasets, the bus and background are not separated even when the cluster number rises to 10 (see Fig. 5).

We consider Q as soft pairwise priors and use the proposed MCNC-PP-(b) algorithm to combine the bottom up segmentation results with the motion priors. There are $\sim 10^4$ prior columns, and it would be too time consuming if we took all prior columns. To reduce computational time, the trajectories are over-segmented into 20 segments and 5 trajectories are randomly selected from each segment. See Fig. 6 for the columns of Q (shown as grayscale images) corresponding to three selected trajectories.

B. Object Segmentation by MCNC With Spatial Regularity

Applying the proposed MCNC algorithms on the priors, we could output X^* (for Algorithm 1) or $\widetilde{X^*}$ (for Algorithm 2) which indicates "soft" segments (see Fig. 7 for the visualization of X^* of an example). To achieve "hard" segmentation,

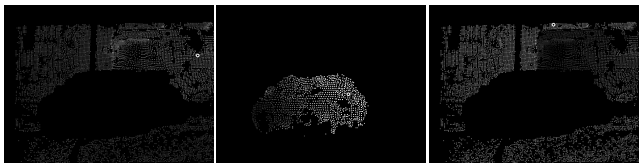


Fig. 6. Three columns selected from the prior matrix (the reference point is marked as a white circle, and its associated pairwise constraints are represented by the bright points with brightness indicating the strength) for *cars10* video in Hopkins155 datasets.

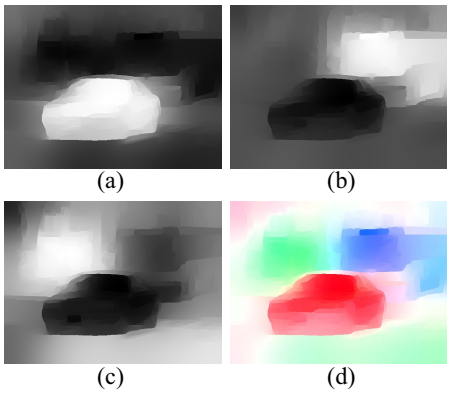


Fig. 7. Visualization of $\widetilde{X^*}$. (a)-(c) 3 columns of the $\widetilde{X^*}$ by using Algorithm 2 on *cars10* video; (d) "soft" segmentation results with channel k corresponding to $(\max d) - d_k$, where d is calculated from $\widetilde{X^*}$ by eq. (13).

one could use k-means or the approaches listed in Algorithms 1–2. However, since these approaches approximate X^* (or $\widetilde{X^*}$) by multiple constant functions, they may produce segments with edges locating in the smooth areas, which are undesirable.

We propose to minimize an energy function that comprises a unary cost term J_{u} representing the total distance between the points and their corresponding segments and a pairwise term J_{p} encoding the spatial non-smoothness:

$$\min_{\pi} \sum_{k=1}^{c} \sum_{i=1}^{n} \delta(k, \pi_i) d_k(i) + \gamma \sum_{i=1}^{n} \sum_{j \in \mathcal{N}(i)}^{n} \delta(\pi_i, \pi_j) d(i, j), \tag{12}$$

where $\pi \in \{1, \ldots c\}$ is the assignment function; δ is a Dirac delta function; $\mathcal{N}(i)$ indicates the neighboring node set (4-neighborhood is adopted); $d_k(i)$ is a unary potential function indicating the cost of node i belonging to segment k; $d(i, j)$ is a pairwise potential function indicating the cost of node i and node j belonging to a same segment; and γ is a parameter that steers the tradeoff between the two terms J_{u} and J_{p}.

We define the unary and pairwise potential functions as (for pairwise priors, X^* is replaced by $\widetilde{X^*}$)

$$d_k(i) = \max(X^*_{i,:}) - X^*_{i,k}, \tag{13}$$

$$d(i, j) = \frac{1}{\|X^*_{i,:} - X^*_{j,:}\|_2^{2\eta}}, \tag{14}$$

where the parameter η is used to control the penalty gap between large and small X^* differences. In the experiments, we fix $\eta = 2$.

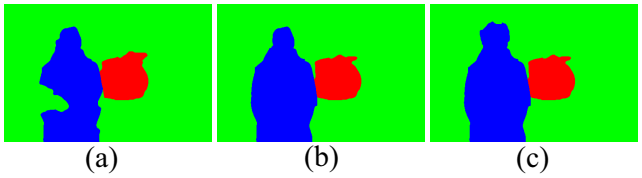


Fig. 8. Segmentation results for the image and priors in Fig. 4 by using different objective functions. (a) segmentation results by directly using the approach in Algorithm 1; (b) segmentation results using the MRF framework but without penalty term J_{l1}; (c) segmentation results by minimizing $J_{\mathrm{interaction}}$ in eq. (21).

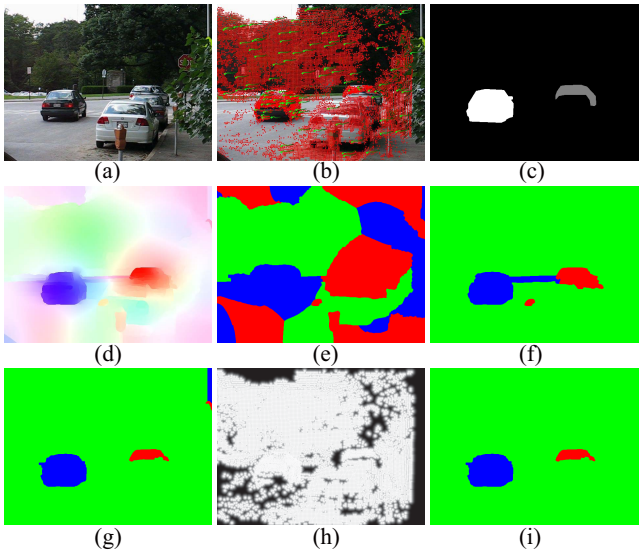


Fig. 9. Segmentation results obtained by using different objective functions. (a) One frame from *cars2* video of Hopkins155 dataset; (b) extracted trajectories; (c) ground truth segments; (d) "soft" segmentation result; (e) segmentation result by directly using scheme (b) in Algorithm 2; (f) segmentation result using the MRF framework but without penalty term J_{l2}; (g) segmentation result without using the weight map WM; (h) the penalty weight map; (i) segmentation result by minimizing J_{motion} in eq. (23).

The optimization of eq. (12) can be solved by any Markov Random Field (MRF) solver. We use the Primal-Dual solver introduced in [24], [25] due to its high efficiency and accuracy.

For the applications which utilize priors from user interaction or motion analysis, the priors are much more reliable. Thus we further add a penalty term to reinforce that the labeled points or extracted trajectories should be assigned with correct labels. We set the penalty factor as 10^4. See Fig. 8–9 for the improvements due to this term.

For interactive segmentation, the penalty term is

$$J_{\mathrm{l1}} = \sum_{k=1}^{c} \sum_{V_i \in C_k^p} \delta(k, \pi_i). \tag{15}$$

For motion guided object segmentation, the penalty term is

$$J_{\mathrm{l2}} = \sum_{k=1}^{c} \sum_{V_i \in C^p} \delta(k, \pi_i) \widehat{d}_k(i), \tag{16}$$

where $\widehat{d}_k(i)$ is computed as follows: using scheme (b) in Algorithm 2, we get the average motion vector for the k^{th} prior cluster as $\overline{\mathbf{u}_k}$; then we calculate the probability of a prior

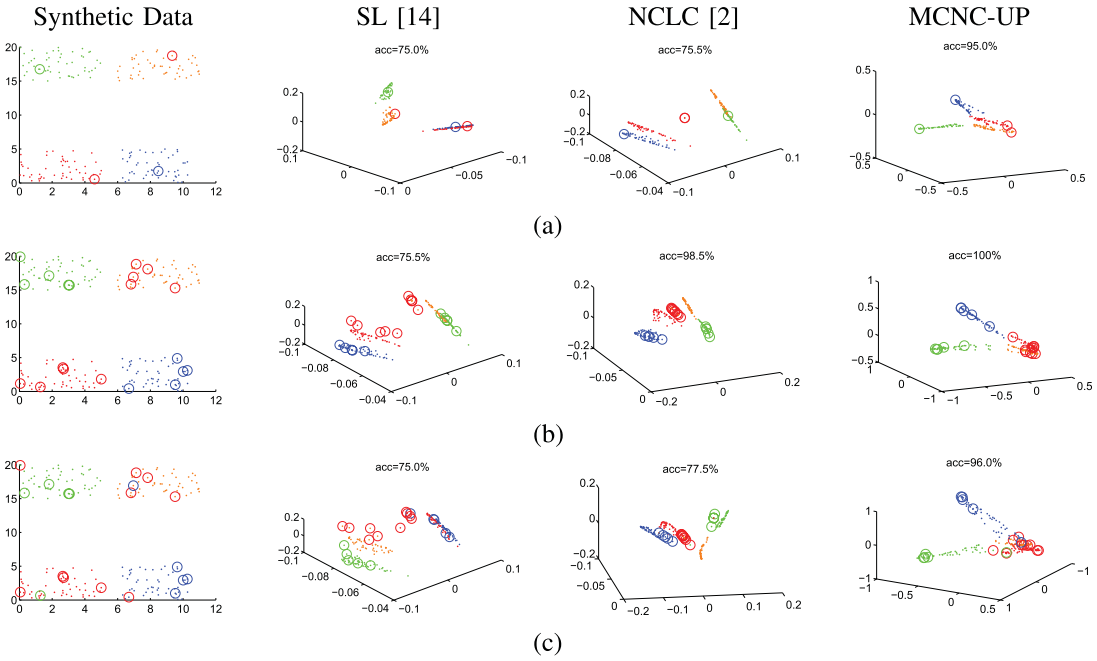


Fig. 10. Comparison of the proposed MCNC-UP algorithm to two related algorithms: SL [14] and NCLC [2]. There are 4 groups of data points following uniform distributions of different ranges, G_1, G_2, G_3, G_4, colored by red, blue, green and orange, respectively. Suppose G_1 and G_4 are from a same cluster and we have obtained the labels for several points (shown as circle points) in each group. The spectral embedding results as well as the accuracies of all algorithms indicate that the proposed MCNC-UP algorithm performs much better than SL and NCLC, especially when the number of labeled points is small or outliers exist. (a) 1 labeled point for each group. (b) 5 labeled points for each group and (c) 5 labeled points but 1 outlier for each group.

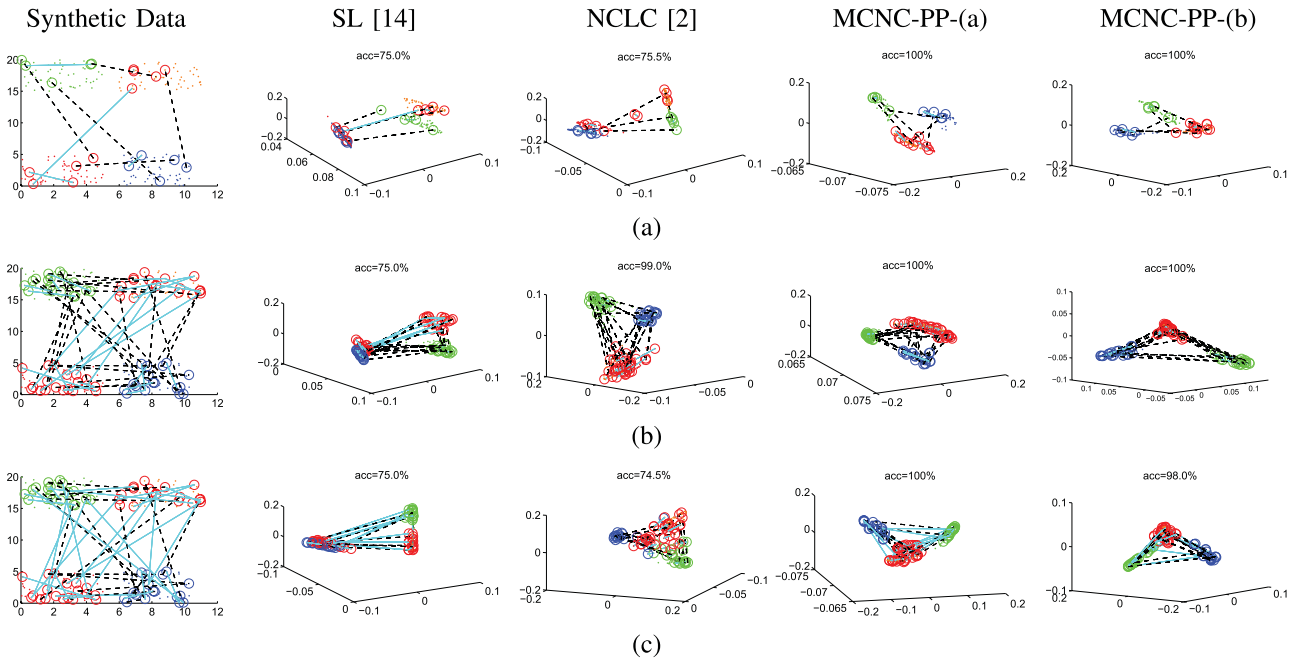


Fig. 11. Comparison of the proposed MCNC-PP algorithms to two related algorithms: SL [14] and NCLC [2]. We use the same data points and graph as in Fig. 10 and generate several must-link (green solid lines with two circles on the two ends) and cannot-link (black dashed lines) constraints. The spectral embedding results as well as the accuracies of all algorithms indicate that both variants of the proposed MCNC-PP algorithm perform much better than SL and NCLC, especially when the number of labeled points is small or outliers exist. (a) 1 pairwise constraint for each group pair. (b) 5 pairwise constraints for each group pair and (c) 5 pairwise constraints for each group pair but with 20% outliers.

point V_i belonging to cluster k as

$$\widehat{A}_{ik} = \exp\left(-\frac{2\|\mathbf{u}_i - \overline{\mathbf{u}_k}\|_2^2}{\overline{\sigma_k}^2}\right), \tag{17}$$

where $\overline{\sigma_k}$ denoting the standard deviation of all distances associated with $\overline{\mathbf{u}_k}$; finally, we compute $\widehat{d}_k(i)$ as

$$\widehat{d}_k(i) = \max(\widehat{A}_{i:}) - \widehat{A}_{ik}. \tag{18}$$

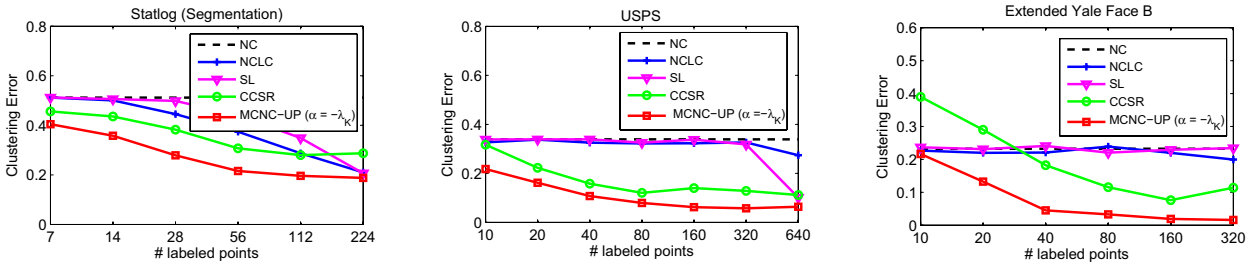


Fig. 12. Comparison of MCNC-UP algorithm to the state-of-the-art algorithms (*Clustering Error vs. # Labeled Points*) on three datasets.

There may still be some errors in the areas without any trajectory points. We further add a weight WM to the unary term in eq. (12) as

$$J_{\text{wu}} = \sum_{k=1}^{n} \sum_{i=1} WM_i \delta(k, \pi_i) d_k(i), \tag{19}$$

where WM is defined based on the distance transform DT [26] of the prior map as

$$WM_i = \exp\left(-\frac{DT_i^2}{2\overline{DT^2}}\right), \tag{20}$$

with $\overline{DT}$ denoting the average distance. In this way, small unary penalty is put on the pixels with no prior points around, and these pixels are more probably assigned with the same labels as their neighborhoods (see Fig. 9).

The final objective functions for the three types of priors are summarized as follows.

Interaction priors:

$$J_{\text{interaction}} = J_{\text{u}} + \gamma J_{\text{p}} + \mu J_{\text{l1}}. \tag{21}$$

Object detection priors:

$$J_{\text{objdet}} = J_{\text{u}} + \gamma J_{\text{p}}. \tag{22}$$

Motion pairwise priors:

$$J_{\text{motion}} = J_{\text{wu}} + \gamma J_{\text{p}} + \mu J_{\text{l2}}. \tag{23}$$

VI. Experiments

A. Experiments on Synthetic Data

We first illustrate the proposed MCNC-UP and MCNC-PP algorithms using synthetic data. The results of two related algorithms, Spectral Learning (SL) [14], Normalized Cut with Linear Constraints (NCLC) [2], are also shown.

Consider the data points in the 1st column of Fig. 10. We construct a RBF-kernel graph with $\sigma = 3$. Suppose the red and orange points are from a same cluster, and we have obtained several labeled points (marked as colored circles). For MCNC-UP algorithm, we set $K = 10$ and $\alpha = -\lambda_K$ and draw the 3D visualizations of X^* obtained by the proposed MCNC-UP algorithm (see the 4th column). For the two related algorithms, we draw their embedded spectrums (see the 2nd and 3rd columns).

When the number of constraints is small (see Fig. 10(a)), SL and NCLC algorithms fail to separate G_1, G_2 or G_3, G_4, or merge G_1, G_4 while the proposed MCNC-UP algorithm succeeds, indicating that the proposed algorithm makes a better encoding of the priors. When the number of constraints increases, the clustering results of NCLC and the MCNC-UP algorithms become comparable (see Fig. 10(b)). When there exist outliers, the proposed algorithm outperforms NCLC algorithm significantly (see Fig. 10(c)), demonstrating the advantage of our soft encoding over the hard one of NCLC in handling noisy priors.

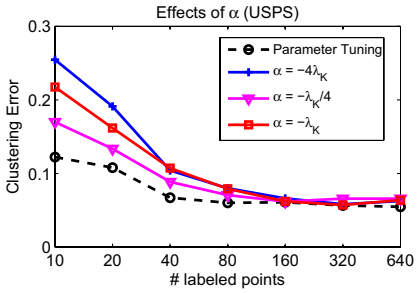


Fig. 13. Effects of α in the proposed MCNC-UP algorithm on USPS datasets.

We also conduct experiments using pairwise priors. We use the same data points and graph as in unary-prior experiments, and randomly generate pairwise constraints for each group pair. The embedded spectrums of all algorithms with varying constraint numbers are shown in Fig. 11. We can see that both the two variants of MCNC-PP algorithm work well in all cases, while SL fails in all cases and NCLC algorithm fails when the number of constraints is small or outliers exist.

The performance of the method in [19] is not included in these figures since its results are difficult to visualize. We will do comparisons with it using real data in the next section.

B. Quantitative Evaluation on Public Datasets

In this section, we compare the proposed algorithms to the state-of-the-art algorithms: Spectral Learning (SL) [14], NCLC [2], and Constrained Clustering with Spectral Regularization (CCSR) [19] on real data. All the algorithms can handle pairwise priors, and thus can also utilize unary priors (pairwise priors can be deduced from unary ones). The results of Normalized Cut (NC) [11] are also shown for reference.

We use three public datasets, Statlog (Segmentation), USPS and Extended Yale Face B, for quantitative evaluation. Statlog (Segmentation) [27] is an image segmentation dataset with

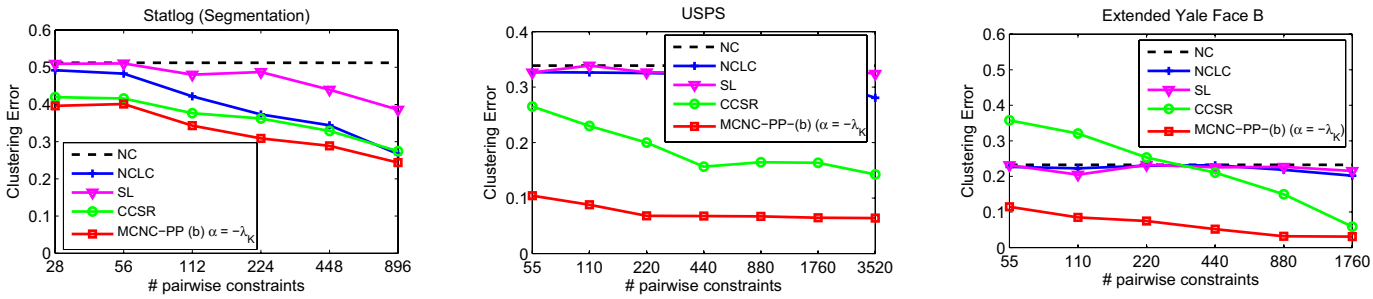


Fig. 14. Comparison of the proposed MCNC-PP algorithm to the state-of-the-art algorithms with pairwise priors (*Clustering Error vs. # Pairwise Constraints*) on three datasets.

TABLE I

COMPUTATIONAL TIMES FOR STATLOG (SEGMENTATION) DATASET ON A COMPUTER WITH 2.4 GHz CPU USING MATLAB 2010A

	Computational Times for Unary Priors (s)				Computational Times for Pairwise Priors (s)				
# constraints	NCLC	SL	CCSR	**MCNC-UP**	NCLC	SL	CCSR	**MCNC-PP (a)**	**MCNC-PP (b)**
7 (28)	3.1	0.7	39.7	**0.04**	2.3	0.8	37.6	**0.8**	**0.2**
28 (112)	3.2	0.7	43.9	**0.05**	2.5	0.9	38.9	**1.6**	**0.5**
224 (896)	19.1	0.8	55.0	**0.06**	4.5	0.9	40.5	**4.7**	**2.8**

2086 non-redundant pixels from 7 outdoor images. A feature with 19 elements is extracted for each pixel and the task is to cluster pixels into 7 classes: brickface, sky, foliage, cement, window, path and grass. USPS[2] is a handwritten digits dataset with 9298 images. For this dataset, an image is represented by a 256-D feature vector formed by concatenating all the columns of the image intensities and the task is to cluster the images into 10 classes (i.e., 10 digits). Extended Yale Face B dataset [28] contains 16128 images of 28 human subjects under 9 poses and 64 illumination conditions. We resize the images with 30×40 pixels, and choose the last 10 subjects (5760 images) for experiments. The goal is to cluster the face images into 10 human subjects.

To make fair comparisons, we use the same graphs for all algorithms. For Statlog (Segmentation) and USPS datasets, We use the weighted k-nearest-neighbor RBF-kernel graph with $k = 20$ and σ determined following the self-tuning algorithm [29]. For Extended Yale Face B dataset, we use a low-rank subspace kernel [30] to construct the graph:

$$W_{ij} = \left(\frac{V_{Di} V_{Dj}^T}{\|V_{Di}\| \|V_{Dj}\|} \right)^2, \tag{24}$$

where V is from SVD factorization of the original data matrix $M = U \Sigma V^T$, and V_{Di} is the i^{th} row of V_D, the first d columns of V (we set $d = 20$ in the experiments). For the proposed MCNC-UP and MCNC-PP algorithms, we fix the parameters in all experiments as: $K = 20$, $\alpha = -\lambda_K$. For all the other algorithms, we use the settings advised by the authors.

For each dataset, several different numbers of labeled points or pairwise constraints are randomly generated using ground truth labels. For a fixed number of constraints, we report the results averaged over 10 trials.

[2] *http://www-i6.informatik.rwth-aachen.de/~keysers/usps.html*

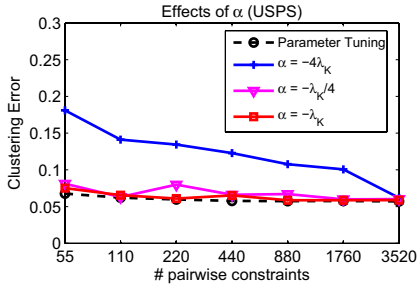


Fig. 15. Effects of α in the proposed MCNC-PP algorithm on the USPS dataset.

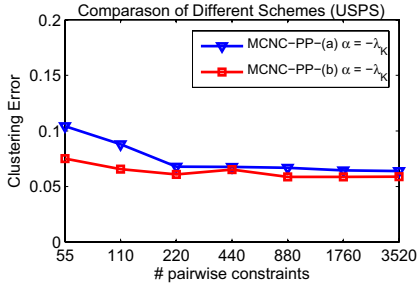


Fig. 16. Comparison of two schemes for MCNC-PP algorithm on the USPS dataset.

We use clustering error (ERR) as the evaluation metric. Denote q_i as the clustering label of x_i obtained by the clustering algorithm and p_i as the ground truth label of $\mathbf{x}_i$. ERR is defined as:

$$ERR = 1 - \frac{1}{n} \sum_{i=1}^{n} \delta(p_i, \text{map}(q_i)), \tag{25}$$

where n is the number of samples, $\delta(x, y) = 1$ if $x = y$; $\delta(x, y) = 0$ otherwise, and $\text{map}(q_i)$ is the best mapping function that permutes clustering labels to match the ground truth labels using the Kuhn-Munkres algorithm.

Fig. 17. Interactive image segmentation. From left to right: original image and labeled points, bottom up segmentation at two scales, segmentation results of NCLC [2], segmentation results of CCSR [19], our "soft" segmentation results, and our segmentation results.

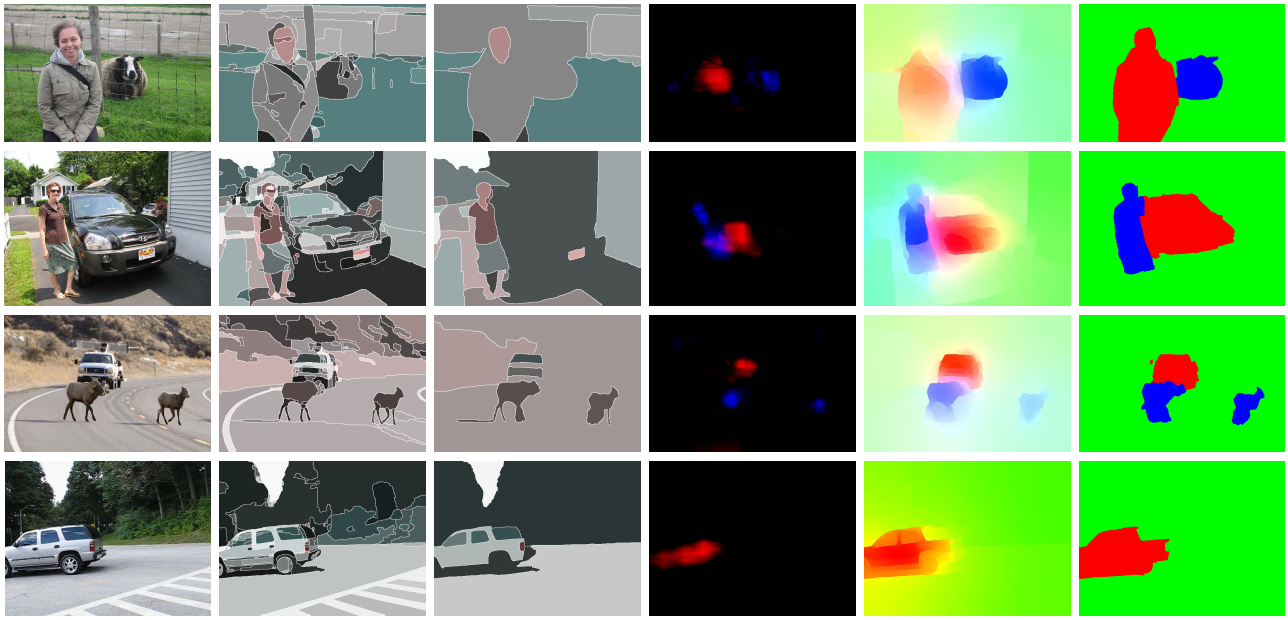


Fig. 18. Constrained image segmentation using object detection priors. From left to right: original image, bottom up segmentation at two scales, soft priors from object detectors, our "soft" segmentation results, and our segmentation results.

Accuracies With Unary Priors: Fig. 12 shows *Clustering Error vs. # Labeled Points* on the three datasets for all algorithms. We can see that the proposed MCNC-UP performs much better than all the other algorithms. We also evaluate the effects of α[3] (see Fig. 13). We can find that when the number of labeled points is small, a larger α yields better results; while when there are a large number of labeled points, a smaller α is better. Since α balances the original normalized cut objective and the correlation penalty, when more labeled points are available, higher confidence should be given to the correlation penalty. This observation provides a guiding principle for parameter selection.

Accuracies With Pairwise Priors: Fig. 14 shows *Clustering Error vs. # Pairwise Constraints* on the three datasets for all algorithms. We can see that the proposed MCNC-PP also performs much better than all the other algorithms. In Fig. 15, we draw the effects of α which also advocates that a smaller α would generate better results when the number of constraints is small. We also compare two schemes of MCNC-PP algorithm (see Fig. 16). We can see that their accuracies are similar especially when the number of constraints is large.

Computational Times: The computational times with varying number of constraints for Statlog (Segmentation) dataset are

[3]*Parameter Tuning* is drawn by tuning α using a set $\text{linspace}(-\lambda_K/4, -\lambda_K, 4) \bigcup \text{linspace}(-\lambda_K, -4\lambda_K, 4)$.

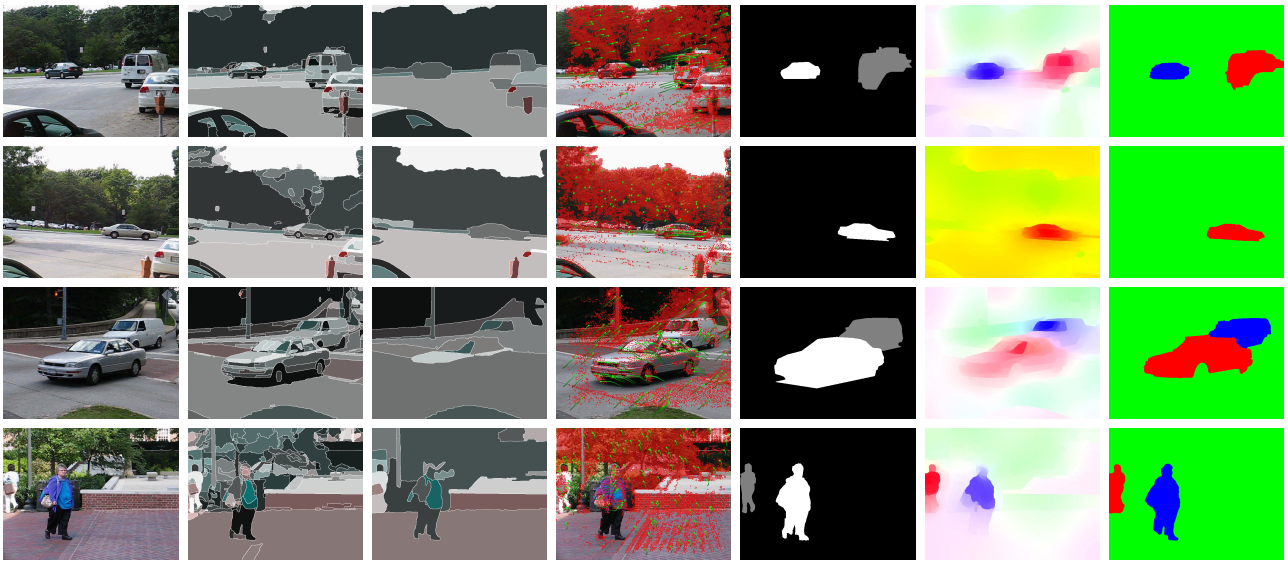


Fig. 19. Constrained image segmentation using motion relations. From left to right: original image, bottom up segmentation at two scales, trajectory points, ground-truth segments, our "soft" segmentation results, and our segmentation results.

shown in Table I. For CCSR and the proposed algorithms, we suppose the first K eigenvectors has been computed (about 1 second using our computer). We can see from Table I that our algorithms are faster than most of the other ones, especially in the case of unary priors. We also compare two variants of the MCNC-PP algorithm and find that scheme (b) is much faster.

C. Experiments on Object Segmentation

We conduct experiments on object segmentation by utilizing three different types of priors as presented in Section V. We use the images from PASCAL VOC datasets [31] and Hopkins155 video datasets [32]. For all images, the method of [1] is used to construct the weighted graph and $K = 31$ eigenvectors are pre-computed. We set $\alpha = -\lambda_K, 0, -5\lambda_K$ for interactive segmentation (reliable priors but in small amount), object detection guided segmentation (less reliable priors) and motion guided segmentation (reliable and dense priors), respectively, according to the reliability and amount of priors. We set $\gamma = \frac{10^{-3}}{n^2\sqrt{n}}$ for all the three applications. Since previous algorithms either cannot handle soft priors or support only two-class problems, they are only tested in interaction cases.

Interactive Segmentation: Fig. 17 shows several examples of interactive segmentation. We can see that the proposed method works very well in all examples, while the two related methods fail. In addition, the segmentation takes only 0.4 seconds for most interactions[4], with about 0.3 seconds to get soft segments and about 0.1 seconds to update the segments, which is sufficiently fast for real-time interaction. Note that the eigenvectors in bottom up segmentation can be precomputed and an initial segmentation is obtained in the first interaction.

Segmentation Using Object Detection Priors: Several examples are shown in Fig. 18. The proposed method can separate the foregrounds from backgrounds, although some errors exist in border areas where few priors are present.

[4]Since the eigenvector computation is only related to the image itself and can be pre-computed off-line, we do not include such time.

Segmentation Using Pairwise Motion Relations: Some examples can be found in Fig. 19. The segmentation results are very close to the ground truths, demonstrating the effectiveness of our method.

VII. Conclusion

In this paper, we have proposed novel algorithms for constrained normalized cut problem, which handle both hard and soft, both unary and pairwise priors in multi-class settings. We also applied the proposed algorithms to object segmentation problem by utilizing priors from human interaction, object detection, and trajectory-based motion analysis.

The major difficulties for utilizing various priors in constrained normalized cut problem lie in two aspects: non-transitive nature of cannot-link constraints and inconsistent representations between given priors and output labels. We settled the difficulties in two novel ways: 1) we incorporated priors into the normalized cut problem using a correlation function. Compared with most previous methods which are limited to hard constraints and impose penalties only on points with constraints, the correlation function makes soft and cannot-link priors tractable, and have the merit of penalizing all points instead of isolated ones; 2) we handled pairwise priors by decomposing the problem into several 2-class unary-prior-based constrained normalized cut subproblems. In this way, the matrix-form priors and vector-form labels are made consistent with each other such that they can be modeled by correlation functions.

We provided closed form solutions for the proposed optimizations and showed that fast computations are achievable for interactive applications. We also applied the proposed algorithms to object segmentation problem by further introducing a spatial regularity term. Extensive experiments were conducted on synthetic data, real image data and object segmentation datasets, proving that the proposed algorithms outperform the state-of-the-art ones significantly in accuracy and also are much faster in most cases.

Our future works will mainly focus on generating more powerful priors to achieve object segmentation. For example, in object detection guided segmentation applications, presently we generate priors directly from responses of sliding window based object detectors, which cover only the center areas of objects. In the future, we will attempt to generate better priors by utilizing more sophisticated object models that contain exact region information [3], [33], [34].

APPENDIX A
PROOF OF THEOREM 1

We first rewrite the constraints in eq. (7) as,

$$\mathrm{diag}(X^T X) = \mathrm{diag}(I) \Rightarrow \mathrm{tr}(X^T X - I) = 0, \tag{26}$$

$$X^T D^{1/2}\mathbf{1} = \mathbf{0} \Leftrightarrow \mathrm{tr}(X^T D^{1/2}\mathbf{1}\mathbf{1}^T D^{1/2} X) = 0, \tag{27}$$

$$\begin{aligned} &\mathrm{diag}(X^T S) \geq \kappa \mathrm{diag}(I) \\ \Leftrightarrow\ &\forall \text{ diagonal } B \geq 0,\ \mathrm{tr}(X^T S B) \geq \kappa \mathrm{tr}(B). \end{aligned} \tag{28}$$

From eq. (26–27), we get

$$\mathrm{tr}(X^T (I - D^{1/2}\mathbf{1}\mathbf{1}^T D^{1/2}) X - I) = 0. \tag{29}$$

Denote $L_{\mathrm{n}} = I - D^{1/2}\mathbf{1}\mathbf{1}^T D^{1/2}$. Replacing the constraints in eq. (7) by eq. (28–29), we get

$$\begin{aligned} &\min_{X}\ \mathrm{tr}(X^T L_{\mathrm{sym}} X) \\ &s.t.\ \ \mathrm{tr}(X^T L_{\mathrm{n}} X - I) = 0, \mathrm{tr}(X^T S B) \geq \kappa \mathrm{tr}(B). \end{aligned} \tag{30}$$

Using Schur Complement [35], we get the dual problem of eq. (30) as

$$\begin{aligned} &\max_{\alpha,\beta,\Lambda}\ \mathrm{tr}(\Lambda) \\ &s.t.\quad \beta \geq 0, \Lambda \text{ diagonal}, \\ &\qquad \begin{bmatrix} L_{\mathrm{sym}} - \alpha L_{\mathrm{n}} & -\beta S B \\ -\beta (S B)^T & \alpha I + \beta \kappa B - \Lambda \end{bmatrix} \succeq 0. \end{aligned} \tag{31}$$

Eq. (31) is a semidefinite programming (SDP) problem, and its dual is

$$\begin{aligned} &\min_{X,Z}\ \mathrm{tr}(L_{\mathrm{sym}} Z) \\ &s.t.\ \ \mathrm{tr}(X^T S B) \geq \kappa \mathrm{tr}(B), \\ &\qquad \mathrm{tr}(L_{\mathrm{n}} Z - I) = 0, \begin{bmatrix} Z & X \\ X^T & I \end{bmatrix} \succeq 0. \end{aligned} \tag{32}$$

For the above equations, we have two propositions as 1 and 2. Then the optimal points of eq. (30), X^*, and the corresponding optimal dual points (α^*, β^*) must satisfy the Karush-Kuhn-Tucker conditions [35],

$$\begin{cases} L_{\mathrm{sym}} X^* - \alpha^* L_{\mathrm{n}} X^* - \beta^* S B = 0, \\ \mathrm{tr}(X^{*T} L_{\mathrm{n}} X^* - I) = 0, \mathrm{tr}(X^{*T} S B - \kappa B) \geq 0, \\ L_{\mathrm{sym}} - \alpha^* L_{\mathrm{n}} \succeq 0, \beta^* \geq 0, \\ \alpha^* \mathrm{tr}(X^{*T} L_{\mathrm{n}} X^* - I) = 0, \beta^* \mathrm{tr}(X^{*T} S B - \kappa B) = 0. \end{cases} \tag{33}$$

Together with the original constraint $X^T D^{1/2}\mathbf{1} = \mathbf{0}$ in eq. (7), X^* is given by $X^* = \beta (L_{\mathrm{sym}} - \alpha L_{\mathrm{n}})^{\dagger} S B$, where $\alpha \leq \lambda_2, \beta > 0$ and B are the optimal solutions of the dual problem eq. (31). Since in eq. (31), βB can be determined by α through constraint $diag(X^T X - I) = 0$ (the left part of eq. (26)) and the facilitate variable Λ (just used to rewrite the original dual problem as a standard SDP problem) could be eliminated, α can be regarded as a function of κ. □

Proposition 1. Strong duality holds between SDPs (31) and (32) for $0 < \kappa < 1$.

Proof: SDP (32) is strictly feasible when $0 < \kappa < 1$ (consider $\widehat{X} = S$. We have $\mathrm{tr}(\widehat{X}^T S B) = \mathrm{tr}(B) > \kappa \mathrm{tr}(B)$, and there certainly exists $\widehat{Z}$ satisfying all the equality and strict inequality constraints). According to Slater's condition [35], strong duality holds between SDPs (31) and (32). □

Proposition 2. Strong duality holds between eq. (30) and SDP (31).

Proof: Eq. (30) is equivalent to

$$\begin{aligned} &\min_{X,Z}\ \mathrm{tr}(L_{\mathrm{sym}} Z) \\ &s.t.\ \ \mathrm{tr}(L_{\mathrm{n}} Z - I) = 0, \mathrm{tr}(X^T S B) \geq \kappa \mathrm{tr}(B), Z = X X^T. \end{aligned} \tag{34}$$

We can find that SDP (32) is a relaxation of eq. (34), and it is obvious that the optimal value of eq. (32) is less than or equal to the optimal value of eq. (34). Together with Proposition 1, we conclude that strong duality between eq. (30) and SDP (31) also holds. □

REFERENCES

[1] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.

[2] S. X. Yu and J. Shi, "Segmentation given partial grouping constraints," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 173–183, Feb. 2004.

[3] M. P. Kumar, P. H. S. Torr, and A. Zisserman, "OBJ cut," in *Proc. IEEE Conf. CVPR*, vol. 1. Jun. 2005, pp. 18–25.

[4] T. Brox and J. Malik, "Object segmentation by long term analysis of point trajectories," in *Proc. ECCV*, vol. 5. 2010, pp. 282–295.

[5] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut—Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.

[6] B. Ghanem and N. Ahuja, "Dinkelbach NCUT: An efficient framework for solving normalized cuts problems with priors and convex constraints," *Int. J. Comput. Vis.*, vol. 89, no. 1, pp. 40–55, 2010.

[7] N. Sundaram, T. Brox, and K. Keutzer, "Dense point trajectories by GPU-accelerated large displacement optical flow," in *Proc. ECCV*, vol. 1. 2010, pp. 438–451.

[8] Y. Ostrovsky, E. Meyers, S. Ganesh, U. Mathur, and P. Sinha, "Visual parsing after recovery from blindness," *Psychol. Sci.*, vol. 20, no. 12, pp. 1484–1491, 2009.

[9] R. Vidal, "Subspace clustering," *IEEE Signal Process. Mag.*, vol. 28, no. 2, pp. 52–68, Mar. 2011.

[10] L. S. da Silva and J. Scharcanski, "Video segmentation based on motion coherence of particles in a video sequence," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 1036–1049, Apr. 2010.

[11] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[12] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.

[13] U. von Luxburg, "A tutorial on spectral clustering," *Stat. Comput.*, vol. 17, no. 4, pp. 395–416, 2007.

[14] S. D. Kamvar, D. Klein, and C. D. Manning, "Spectral learning," in *Proc. IJCAI*, 2003, pp. 561–566.

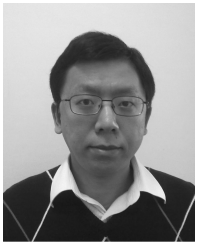
[15] B. Kulis, S. Basu, I. S. Dhillon, and R. J. Mooney, "Semi-supervised graph clustering: A kernel approach," in *Proc. 22nd ICML*, 2005, pp. 457–464.

[16] Z. Lu and M. Á. Carreira-Perpiñán, "Constrained spectral clustering through affinity propagation," in *Proc. IEEE Conf. CVPR*, Jun. 2008, pp. 1–8.

[17] A. P. Eriksson, C. Olsson, and F. Kahl, "Normalized cuts revisited: A reformulation for segmentation with linear grouping constraints," in *Proc. IEEE 11th ICCV*, Oct. 2007, pp. 1–8.

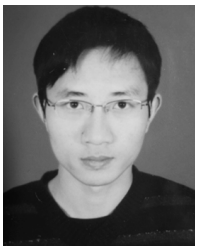
[18] L. Xu, W. Li, and D. Schuurmans, "Fast normalized cut with linear constraints," in *Proc. IEEE Conf. CVPR*, Jun. 2009, pp. 2866–2873.
[19] Z. Li, J. Liu, and X. Tang, "Constrained clustering via spectral regularization," in *Proc. CVPR*, 2009, pp. 421–428.
[20] X. Wang and I. Davidson, "Flexible constrained spectral clustering," in *Proc. 16th ACM SIGKDD Int. Conf. KDD*, 2010, pp. 563–572.
[21] S. Maji, N. K. Vishnoi, and J. Malik, "Biased normalized cuts," in *Proc. CVPR*, 2011, pp. 2057–2064.
[22] H. Hu, J. Zhou, J. Feng, and J. Zhou, "Multi-way constrained spectral clustering by nonnegative restriction," in *Proc. ICPR*, 2012, pp. 1550–1553.
[23] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed. Cambridge, MA, USA: MIT Press, 2001.
[24] N. Komodakis and G. Tziritas, "Approximate labeling via graph cuts based on linear programming," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 8, pp. 1436–1453, Aug. 2007.
[25] N. Komodakis, G. Tziritas, and N. Paragios, "Performance vs computational efficiency for optimizing single and dynamic MRFs: Setting the state of the art with primal-dual strategies," *Comput. Vis. Image Understand.*, vol. 112, no. 1, pp. 14–29, 2008.
[26] A. Rosenfeld and J. L. Pfaltz, "Sequential operations in digital picture processing," *J. ACM*, vol. 13, no. 4, pp. 471–494, 1966.
[27] A. Frank and A. Asuncion. (2010). *UCI Machine Learning Repository* [Online]. Available: http://archive.ics.uci.edu/ml
[28] K. Lee, J. Ho, and D. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 684–698, May 2005.
[29] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Advances in Neural Information Processing Systems*. Cambridge, MA, USA: MIT Press, 2004.
[30] F. Lauer and C. Schnorr, "Spectral clustering of linear subspaces for motion segmentation," in *Proc. IEEE 12th ICCV*, Sep./Oct. 2009, pp. 678–685.
[31] M. Everingham, L. J. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010.
[32] R. Tron and R. Vidal, "A benchmark for the comparison of 3-D motion segmentation algorithms," in *Proc. IEEE Conf. CVPR*, Jun. 2007, pp. 1–8.
[33] E. Borenstein and S. Ullman, "Combined top-down/bottom-up segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 12, pp. 2109–2125, Dec. 2008.
[34] J. Carreira and C. Sminchisescu, "CPMC: Automatic object segmentation using constrained parametric min-cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1312–1328, Jul. 2012.
[35] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

Han Hu received the B.S. degree from the Department of Automation, Tsinghua University, Beijing, China, in 2008, where he is currently pursuing the Ph.D. degree with the Department of Automation. His current research interests include industrial image processing, data clustering, and visual segmentation.



Jianjiang Feng (M'10) received the B.S. and Ph.D. degrees from the School of Telecommunication Engineering, Beijing University of Posts and Telecommunications, Beijing, China, in 2000 and 2007, respectively.

He is an Associate Professor with the Department of Automation, Tsinghua University, Beijing. From 2008 to 2009, he was a Post-Doctoral Researcher with the Pattern Recognition and Image Processing Laboratory, Michigan State University, East Lansing, MI, USA. His current research interests include fingerprint recognition, palmprint recognition, and structural matching.



Chuan Yu was born in 1989. He received the B.S. degree from the Department of Automation, Tsinghua University, Beijing, China, in 2012, where he is currently pursuing the M.S. degree with the Department of Automation. His current research interests include computer vision and pattern recognition.

Jie Zhou (M'01–SM'04) was born in 1968. He received the B.S. and M.S. degrees from the Department of Mathematics, Nankai University, Tianjin, China, in 1990 and 1992, respectively, and the Ph.D. degree from the Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology, Wuhan, China, in 1995. In 1997, he served as a Post-Doctoral Fellow with the Department of Automation, Tsinghua University, Beijing, China. Since 2003, he has been a Full Professor with the Department of Automation, Tsinghua University. His current research interests include computer vision, pattern recognition, and image processing. He has authored more than 100 papers in peer-reviewed journals and conferences. He has published more than 30 papers in top journals and conferences, such as IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE and the Conference on Computer Vision and Pattern Recognition. He is an Associate Editor for the *International Journal of Robotics and Automation* and *Acta Automatica*. He is a recipient of the National Outstanding Youth Fund of China.