

PROJECTIVE NON-NEGATIVE MATRIX FACTORIZATION FOR UNSUPERVISED GRAPH CLUSTERING

Christos G. Bampis¹, Petros Maragos^{2,3} and Alan C. Bovik¹

¹Department of E.C.E., University of Texas at Austin, Austin, TX 78712-0240, USA.

²School of E.C.E., National Technical University of Athens, 15773 Athens, Greece.

³Athena Research and Innovation Center, Maroussi 15125, Greece

bampis@utexas.edu, maragos@cs.ntua.gr, bovik@ece.utexas.edu

ABSTRACT

We develop an unsupervised graph clustering and image segmentation algorithm based on non-negative matrix factorization. We consider arbitrarily represented visual signals (in 2D or 3D) and use a graph embedding approach for image or point cloud segmentation. We extend a Projective Non-negative Matrix Factorization variant to include local spatial relationships over the image graph. By using properly defined region features, one can apply our method of unsupervised graph clustering for object and image segmentation. To demonstrate this, we apply our ideas on many graph based segmentation tasks such as 2D pixel and super-pixel segmentation and 3D point cloud segmentation. Finally, we show results comparable to those achieved by the only existing work in pixel based texture segmentation using Nonnegative Matrix Factorization, deploying a simple yet effective extension that is parameter free. We provide a detailed convergence proof of our spatially regularized method and various demonstrations as supplementary material. This novel work brings together graph clustering with image segmentation.

I. INTRODUCTION

The complexity and volume of visual information has led research efforts to efficient methods that can achieve dimensionality reduction while retaining valuable data properties. Non-negative matrix factorization (NMF) techniques [1] have emerged as promising tools for learning parts-based representations of non-negative data. Many variants of NMF methods have been proposed. Ding *et al.* proposed a Convex NMF [2] by restricting the feature basis to a convex combination of data samples and Cluster NMF as its special case. Similar to Cluster NMF, Yuan and Oja proposed Projective NMF (PNMF) [3] which inspired our work and was also used in [4] to construct global low rank similarity matrices for label propagation. Since the k -nn graph can be sensitive to the neighborhood size k , the authors of [4] proposed an approximation method to construct a more robust similarity matrix which essentially uses the same minimization scheme as PNMf. NMF methods have been shown to be equivalent to spectral clustering and probabilistic latent semantic indexing [5], [6]. These methods do not exploit intrinsic information of the original data or prior label information.

In order to address the lack of intrinsic information, graph regularization has also been proposed in [7] (GNMF) and in [1] for document clustering. A random walk approach was proposed in [8] which can take into account further relationships between data samples. A neighborhood preserving PNMf was also suggested in [9] for hyperspectral image classification. Fisher's criterion has been used both for NMF [10] and PNMf [11] in order to

improve classification accuracy through labeled samples. In [12], a semi-supervised version of PNMf was deployed for cancer classification. These previous works clearly demonstrate that PNMf (and NMF) can be used both for supervised and unsupervised tasks. Despite their extensive use in data mining [13], image classification/annotation [14] and other applications, factorization methods have found little application to image and object segmentation. In [15], the authors showed image segmentation results applied on a pixel by pixel basis. To the best of our knowledge, the recent work of Yuan *et al.* [16] is perhaps the only fully developed approach that uses matrix factorization and they focus on the texture segmentation problem. We develop an unsupervised graph clustering scheme and extend it to capture spatial relationships by enforcing local smoothness between nodes. Our method can be readily deployed as a graph based approach or as a pixel based approach which is parameter free and that demonstrates improved results. The rest of this paper is organized as follows. Section II reviews Non-negative Matrix Factorization and describes the original PNMf method. Then, Section III describes the proposed method and in Section IV the pixel and node version of our method is compared to other methods. Section V concludes and discusses future work.

II. NON-NEGATIVE MATRIX FACTORIZATION

Non-negative matrix factorization methods can be used for producing low-rank representations of image data and for clustering. In this work, we exploit the clustering properties of NMF based methods to deploy a graph based clustering approach which can then be used for image and point cloud segmentation. Suppose we are given a $m \times n$ non-negative data matrix \mathbf{X} and an integer $c < \min(n, m)$ where n denotes the number of data samples, m the dimensionality and c is the rank of the desired approximation. The original NMF method [17] decomposes \mathbf{X} into the product of two lower-rank and non-negative matrices: the $m \times c$ basis matrix $\mathbf{W} = [W_{ij}]$ and the $n \times c$ coefficient matrix $\mathbf{H} = [H_{ij}]$ such that $\mathbf{X} \approx \mathbf{W}\mathbf{H}^T$. Since \mathbf{W} and \mathbf{H} have lower rank than \mathbf{X} , the NMF compresses the high dimensional data in \mathbf{X} . From a clustering perspective, \mathbf{W} loosely captures the cluster centroids while \mathbf{H} reveals cluster memberships. The standard NMF is based on minimizing the Euclidean distance between \mathbf{X} and $\hat{\mathbf{X}} = \mathbf{W}\mathbf{H}^T$:

$$\min_{\mathbf{W}, \mathbf{H}} \left\| \mathbf{X} - \mathbf{W}\mathbf{H}^T \right\|^2, \text{ s.t. } \mathbf{W} \geq 0, \mathbf{H} \geq 0 \quad (1)$$

where $\mathbf{W} \geq 0$ and $\mathbf{H} \geq 0$ denotes non-negative matrix elements. Similar optimization problems can be formulated by using other types of distance measures like the Kullback-Leibler divergence [18]. Eq. (1) is non-convex for both \mathbf{W} and \mathbf{H} thus a global solution is not guaranteed. Lee and Seung [17] proposed the following multiplicative update rules (MUR) to solve (1):

$$H_{ij} \leftarrow H_{ij} \frac{(\mathbf{X}^T \mathbf{W})_{ij}}{(\mathbf{H}\mathbf{W}^T \mathbf{W})_{ij}}, \quad W_{ij} \leftarrow W_{ij} \frac{(\mathbf{X}\mathbf{H}^T)_{ij}}{(\mathbf{W}\mathbf{H}\mathbf{H}^T)_{ij}} \quad (2)$$

This work was partially supported by the project COGNIMUSE under the ARISTEIA Action co-funded by the European Social Fund and Greek National Resources. It was also partially supported by the European Union under the projects MOBOT with grant FP7-600796 and DIRHA with grant FP7-288121.

A positive random initialization of \mathbf{W} and \mathbf{H} or other directed initialization method [19], [2] can be used. After computing the basis matrix \mathbf{W} produced by NMF, compute the coefficient of a new example \mathbf{X} by $y = \mathbf{W}^\dagger \mathbf{X}$ where \mathbf{W}^\dagger denotes the pseudo-inverse of \mathbf{W} . This operation might then violate the non-negativity required by NMF, since computing the pseudo-inverse can lead to non-negative entries in \mathbf{W}^\dagger . This phenomenon is known as the “out of sample deficiency” that most NMF-based methods [17], [15] can be sensitive to. In [3], Yuan et al. introduced the Euclidean PNMF method which further constrains the original NMF, i.e.

$$\min_{\mathbf{W}} \left\| \mathbf{X} - \mathbf{W}\mathbf{W}^\top \mathbf{X} \right\|^2 \text{ s.t. } \mathbf{W} \geq 0 \quad (3)$$

For classification, compute the coefficient of a new test sample \mathbf{X} by $y = \mathbf{W}^\top \mathbf{X}$ where y is now non-negative and the “out of sample deficiency” is eliminated. In an unsupervised setting, the PNMF method is closely related to clustering [3] and is equivalent to Cluster NMF [2], [1]. The following MUR was proposed:

$$W_{ij} \leftarrow W_{ij} \frac{(\mathbf{X}\mathbf{X}^\top \mathbf{W})_{ij}}{(\mathbf{W}\mathbf{W}^\top \mathbf{X}\mathbf{X}^\top \mathbf{W})_{ij} + (\mathbf{X}\mathbf{X}^\top \mathbf{W}\mathbf{W}^\top \mathbf{W})_{ij}} \quad (4)$$

where the W_{ij} can be randomly initialized to non-negative values. After every iteration, the original PNMF normalizes \mathbf{W} by its spectral norm $\|\mathbf{W}\|_2$ in order to stabilize the minimization process [3], [18]. This update rule was shown to converge empirically without a theoretical proof. However, even after this normalization, a monotonic decrease in the objective function is not ensured [20]. In [21] the authors suggested a wide range of related methods called Quadratic NMF, which use a learning rate parameter η . A converging MUR using $\eta = \frac{1}{3}$ was proposed for PNMF. The η exponent value influences the convergence [22], i.e. aggressive choices of η can improve the convergence speed but might also endanger the monotonic decrease of the objective function.

III. GRAPH REGULARIZED PROJECTIVE NMF FOR IMAGE/OBJECT SEGMENTATION

III-A. Proposed method using image-driven graphs

We now describe our formulation which uses the reformulated PNMF objective as its data term:

$$J_{\text{data}}(\mathbf{H}) = \left\| \mathbf{X} - \mathbf{X}\mathbf{H}\mathbf{H}^\top \right\|^2 \quad (5)$$

Image regularities are very important in image segmentation applications, therefore spatial regularity should be enforced. Here, we add a spatial constraint to the minimization solution so that neighboring nodes get assigned to the same labels as much as possible. Therefore, we adopt the following minimization scheme:

$$\min_{\mathbf{H}} J(\mathbf{H}), \text{ with } J(\mathbf{H}) = J_{\text{data}}(\mathbf{H}) + \lambda J_{\text{reg}}(\mathbf{H}) \quad (6)$$

with $H_{ij} \geq 0 \forall i, j$, $J_{\text{reg}}(\mathbf{H})$ is the spatial regularizing functional and λ expresses the regularization tradeoff. Large values of λ lead to oversmoothed results while very small values ignore spatial regularities. After computing \mathbf{H} , the label of node i becomes:

$$\text{label}_i = \arg \max_j H_{i,j} \quad (7)$$

Let $G = (V, E)$ be the corresponding graph used in our graph embedding approach. This graph consists of a set of vertices/nodes $v \in V$ and a set of edges $e \in E \subseteq V \times V$ where $|V| = n$. In addition, an edge e_{ij} denotes an edge spanning two vertices v_i and v_j . To model node relationships we use weights encoded in a weight matrix $\mathbf{S} = [S_{ij}]$ and a degree matrix $\mathbf{D} = \text{diag}(d_1, \dots, d_n)$ where $d_i = \sum_{j \sim i} S_{ij}$. We also assume that G is connected and undirected hence \mathbf{S} is symmetric. We use $J_{\text{reg}}(\mathbf{H}) = \text{Tr}(\mathbf{H}^\top \mathbf{L}\mathbf{H})$ [7], where

$\mathbf{L} = \mathbf{D} - \mathbf{S}$ is the standard Graph Laplacian of size $n \times n$. We solve (6) using the following MUR:

$$H_{ij} \leftarrow H_{ij} \sqrt[4]{\frac{[2\mathbf{X}^\top \mathbf{X}\mathbf{H} + \lambda \mathbf{S}\mathbf{H}]_{ij}}{[\mathbf{H}\mathbf{H}^\top \mathbf{X}^\top \mathbf{X}\mathbf{H} + \mathbf{X}^\top \mathbf{X}\mathbf{H}\mathbf{H}^\top \mathbf{H} + \lambda \mathbf{D}\mathbf{H}]_{ij}}} \quad (8)$$

When $\lambda = 0$ this update rule can be considered as a Projective NMF variant. For the derivation and convergence of this update rule see the supplementary material for a complete proof. An outline of our proof can be found in the Appendix.

Our algorithm can be extended from regular grids (see an example in Fig. 1 left) to image-driven graphs (see Fig. 1 right). Similar to [23], we apply a watershed transformation [24] to the original image to obtain a set of n regions R_1, R_2, \dots, R_n . Every region i is associated with a node whose location is the geometric mean \mathbf{x}_i of R_i and whose feature vector is \mathbf{g}_i , e.g. a mean feature vector for all pixels in R_i using color or texture information. Another way to produce these super-pixel regions is the SLIC [25] method. One difference between these two approaches is that SLIC super-pixels are expected to be larger and more visually consistent areas whereas the watershed regions tend to be smaller (due to gradient noise sensitivity) and arbitrarily sized.

Next, we consider only edges between contingent regions, i.e. we use a Region Adjacency Graph (RAG) for G as shown in Fig. 1 right. Therefore, \mathbf{S} is an adjacency matrix, i.e. $S_{ij} = 1$ if $i \sim j$ and $S_{ij} = 0$ else, where \sim denotes that nodes i and j are adjacent. If further knowledge about the problem is available, other weighting schemes can be considered. For example, prior knowledge can be integrated using the normalized Graph Laplacian [26]. Other authors have used a k-nn graph [27] or a locally optimized neighborhood [28] to create an adjacency map between nodes by assuming a fixed number of neighbors k . After obtaining the node labels using (7), we can transform the node solution into a pixel one, i.e. assign all pixels belonging to R_i to their node’s label. We deal with boundary pixels by assigning each one to the most common label present with respect to this pixel’s eight neighbors.

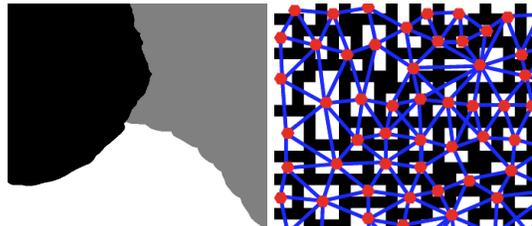


Fig. 1: Left: Example of pixel based result of texture segmentation in the Prague Dataset [16]. Right: An example of the Region Adjacency Graph.

III-B. Comparison with previous works

In [1], it was suggested that the PNMF scheme cannot exploit the intrinsic structure of the data, hence a spatially regularized method was developed for a document clustering task. Different from [1], we propose a fast and efficient extension of PNMF where we factorize only \mathbf{H} and instead use $\eta = 0.25$ to ensure convergence. By adding spatial constraints on the local neighborhood, we capture spatial information and smooth noisy features as part of the image segmentation task. Our theoretical contribution mainly lies in the proof of the correctness and convergence of our method. This result strongly supports the empirical observation of [22] that smaller η values guarantee convergence (using a different mathematical approach). Further, we demonstrate the flexibility and power of this approach in a wide set of graph-related applications.

From a pixel-based perspective, the work of [16] is perhaps the only existing method that uses matrix factorization for pixel image segmentation. The authors of [16] use a factorization based approach comprised of two steps. First, they apply singular value decomposition (SVD) on the data matrix to acquire representative

features [16], reduce the feature dimensionality and automatically detect the number of segments. The next step is to impose non-negativity constraints using NMF by an efficient alternate least squares (ALS) method to minimize the following:

$$f(\mathbf{Z}, \beta) = \|\mathbf{Y} - \mathbf{Z}\beta\|^2 + \lambda_1 \|\mathbf{Z}\|^2 + \lambda_2 \|\beta\|^2 \quad (9)$$

where \mathbf{Y} is the feature matrix and \mathbf{Z} , β are factor matrices. This method produced good results on texture segmentation, but did not generalize well on natural image segmentation since it was based on texture descriptors. A major drawback of this method is that it uses two different matrices for the non-negativity constraint which need to be regularized to avoid large norms in the learned matrices. To tradeoff between the data term and the regularization two different regularization parameters λ_1 and λ_2 were used, which were manually selected to be 0.1. In the next section, we show that our pixel based method delivers better results and is parameter free.

IV. EXPERIMENTAL RESULTS

IV-A. Pixel level image segmentation

In order to demonstrate the flexibility of our method we address a wide range of tasks. First, we consider pixel based texture segmentation. We test our method on the Prague Texture Segmentation Dataset [16] which consists of computer generated texture mosaics and allows for benchmarking results. For fairness between the factorization approach in [16] and our method, we adopt the same strategy for the features and number of segments. We set $\lambda = 0$ in (6) since the SVD based features already capture texture properties and exploit the localization property as suggested in [16]. Table I shows the results using those metrics where the method in [16] (FSeg) performed best compared to other methods. It can be seen that our method delivers better results using most of the reported metrics. We also report results obtained using other possible matrix factorization schemes that could be used in a similar manner. We do not compare with NMF since the other NMF variants have been shown to give better results overall. Using $\lambda = 0$ also means that GNMF [7] is identical to NMF. For all methods 500 iterations were performed to ensure convergence of all of them. It is obvious that Convex NMF performed the worst. Further, our method was insensitive to parameter selection since large matrix norms are naturally eliminated by the η exponent. We call λ_1 and λ_2 in (9) the parameters used for regularizing the norms of the coefficient and the basis matrix in [16] respectively. Fig. 2 shows the sensitivity of FSeg to λ_1 by varying it over the $[0.01 \dots 0.5]$ interval. We report the F-measure computed by the dataset's benchmarking system.

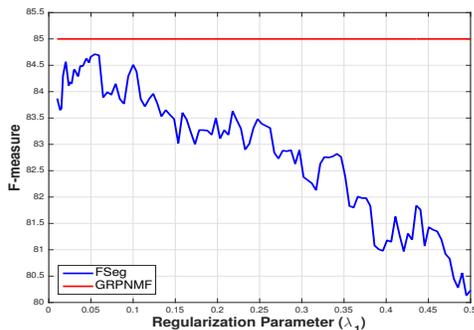


Fig. 2: F-measure against λ_1 values in the Prague Dataset. GRPNMF is independent of λ_1 and outperforms FSeg. This figure is better seen in color.

The benefit of GRPNMF is twofold: the F-measure for GRPNMF is higher across all possible choices of λ_1 without needing such a free parameter. Similar results can be derived for λ_2 although it affects the final result less than λ_1 .

IV-B. Node level image segmentation by graph clustering

Despite the strength of our method in the previous section, our ultimate goal is to apply our algorithm in a graph based approach.

We now describe the results of our node based method on both texture and color segmentation. We used the perceptually uniform Lab colorspace as the color features and Gabor filter responses for texture segmentation [30]. The Gabor filterbank had 5 scales and 8 orientations. For the RAG construction, we applied the watershed transform [24] on the gradient of the grayscale image. First, we compare GRPNMF to NMF and study the effect of regularization. According to Fig. 3, both NMF and GRPNMF with $\lambda = 0$ cannot achieve smooth results. By contrast, using $\lambda > 0$, achieves this spatial regularity and delivers improved results for GNMF and GRPNMF further improves them. By increasing λ , more spatial regularity can be achieved as seen in both the node based clustering and the corresponding pixel image in Fig. 4. Smaller regions are merged with larger ones and region boundaries become smoother. If a very large λ is used, the final solution might be overly smooth.

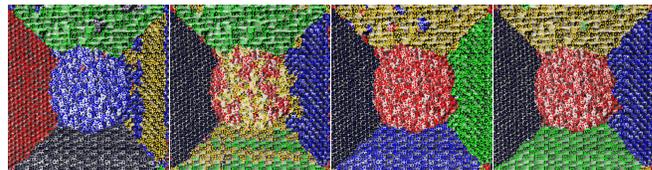


Fig. 3: Left to right: NMF, GRPNMF $\lambda = 0$, GNMF $\lambda = 0.25$, GRPNMF $\lambda = 10$. This figure is better seen in color and zoomed in.

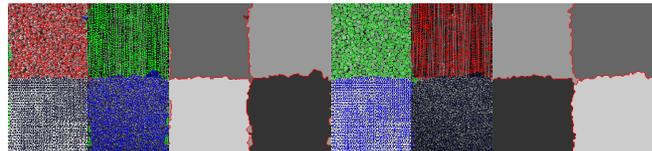


Fig. 4: Effect of regularization (left to right): GRPNMF using $\lambda = 1$, $\lambda = 20$ and pixel texture segmentation with region boundaries in red, 250 iterations. This figure is better seen in color and zoomed in.

We also test our method against other methods for a color based segmentation. According to Fig. 5, GRPNMF outperforms NMF, GNMF and the standard K-means algorithm for both test images. For example, the K-means was prone to the initialization whereas NMF did not capture the objects as good as GRPNMF with regularization. The effect of regularization is also visible: the objects adhere well to their spatial regularities. Fig. 6 also shows the GRPNMF iterations for a test image.

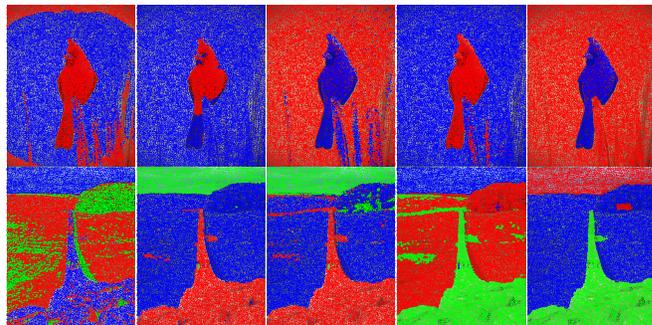


Fig. 5: Natural image segmentation (left to right): K-means, NMF, GRPNMF $\lambda = 0$, GNMF $\lambda = 0.15$, GRPNMF $\lambda = 500$, 2000 iterations for all methods. GNMF tends to be more sensitive to the initializations than GRPNMF. This figure is better seen in color and zoomed in.

IV-C. 3D Point Cloud segmentation

This section considers the problem of segmenting a 3D object given its depth. The Kinect depth camera has made the process of acquiring depth information easier and more consistent. Here, we use GRPNMF to weight the different visual and spatial cues into a common framework which applies graph clustering to object segmentation. Referring to Fig. 7, the depicted object cannot be separated without using the spatial regularizer. Using color (notice

Table I: Pixel Based Texture Segmentation in Prague Dataset. Best result is denoted with bold.

Method	CS(↑)	ME(↓)	NE(↓)	O(↓)	CA(↑)	CO(↑)	I(↓)	EA(↑)	MS(↑)	RM(↓)	CI (↑)	GCE(↓)
FSeg [16]	69.02	6.28	5.66	10.79	77.50	84.11	15.89	83.99	78.25	4.51	84.71	10.82
CNMF [2]	49.32	5.40	5.11	36.81	60.07	70.48	29.52	66.69	57.25	11.01	67.87	9.88
ONMF [29]	68.33	5.81	6.30	10.13	77.20	83.19	16.81	84.23	78.03	4.54	85.18	11.59
GRPNMF (Our)	69.50	5.74	5.89	10.33	77.92	84.00	16.00	84.39	78.57	4.34	85.24	10.61

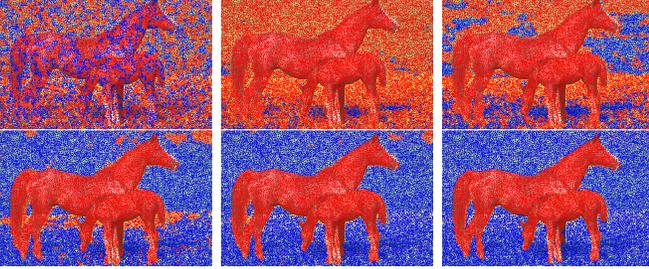


Fig. 6: GRPNMF using $\lambda = 500$: 2, 450, 500, 600, 700 and 1000 iterations. Red and blue denote different clusters. For 1000 iterations and 14459 nodes, GRPNMF took 1.61 sec. on a CPU. This figure is better seen in color.

the shading effect in the floor) or even combining color with space and normal information as features does not produce satisfying results. However, by applying regularized GRPNMF the noisy influence of the normals is smoothed and the object is better separated from the floor. For the graph embedding, we used a k -nn graph using $k = 8$ neighbors.

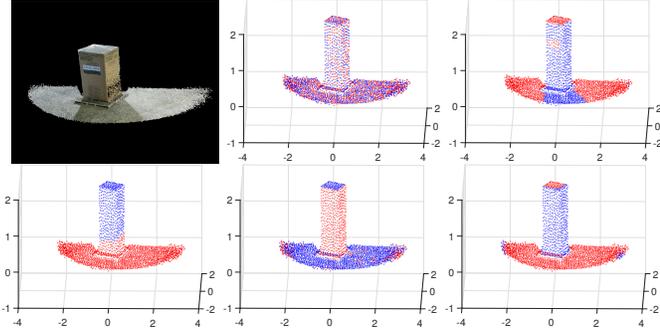


Fig. 7: Effect of using different features on GRPNMF for 3D point cloud segmentation using 2 classes, 5500 iterations (left to right): i) original point cloud, ii) random initialization, iii) color and $\lambda = 0$, iv) space and $\lambda = 0$, v) color, space, normals and $\lambda = 0$, vi) color, space, normals and $\lambda = 500$.

V. CONCLUSION

We developed a graph clustering algorithm based on the Projective non-negative matrix factorization called GRPNMF. Our contributions lie in studying new ways to approach image and object segmentation via unsupervised graph clustering. We improve on a recent factorization based approach for pixel based texture segmentation by proposing a factorization scheme without any regularization parameters. We then present qualitative results on color and texture segmentation problems on image-driven graphs and point cloud segmentation using both color and depth cues, achieving promising results. We provide a theoretical proof for the convergence and correctness of GRPNMF which can be thought of as an extended case of PNMF using $\eta = \frac{1}{4}$. Our future work will be directed to finding better region features for our RAG based graph clustering and further exploitation of matrix factorization techniques for segmentation and object recognition.

VI. APPENDIX

First, we outline the proof of the correctness for update rule (8). For any two matrices \mathbf{X} , \mathbf{H} we have that:

$$\|\mathbf{X} - \mathbf{X}\mathbf{H}\mathbf{H}^\top\|^2 = \text{Tr}\left\{(\mathbf{X} - \mathbf{X}\mathbf{H}\mathbf{H}^\top)^\top(\mathbf{X} - \mathbf{X}\mathbf{H}\mathbf{H}^\top)\right\} \quad (10)$$

where $\text{Tr}\{\cdot\}$ denotes the trace. For the purposes of this section, set $\mathbf{X} \leftarrow \mathbf{X}^\top$. Introduce the Lagrange multipliers by defining a $n \times c$ matrix $\Phi = [\phi_{ij}]$ and then:

$$\min_{\mathbf{H}} \hat{J}(\mathbf{H}), \text{ with } \hat{J}(\mathbf{H}) = J(\mathbf{H}) + \text{Tr}(\Phi\mathbf{H}^\top) \quad (11)$$

Taking the gradient with respect to \mathbf{H} yields:

$$\begin{aligned} \frac{\partial \hat{J}}{\partial \mathbf{H}} &= -2(\mathbf{X} - \mathbf{H}\mathbf{H}^\top\mathbf{X})\mathbf{X}^\top\mathbf{H} - 2\mathbf{X}(\mathbf{X}^\top - \mathbf{X}^\top\mathbf{H}\mathbf{H}^\top)\mathbf{H} \\ &\quad + \lambda(\mathbf{L} + \mathbf{L}^\top)\mathbf{H} + \Phi \\ &= -4\mathbf{X}\mathbf{X}^\top\mathbf{H} + 2\mathbf{H}\mathbf{H}^\top\mathbf{X}\mathbf{X}^\top\mathbf{H} + 2\mathbf{X}\mathbf{X}^\top\mathbf{H}\mathbf{H}^\top\mathbf{H} \\ &\quad + 2\lambda\mathbf{L}\mathbf{H} + \Phi \end{aligned}$$

where we also use that \mathbf{L} is symmetric. By replacing $\mathbf{L} = \mathbf{D} - \mathbf{S}$ and setting the gradient to 0 we get:

$$\begin{aligned} 4\mathbf{X}\mathbf{X}^\top\mathbf{H} + 2\lambda\mathbf{S}\mathbf{H} &= 2\mathbf{H}\mathbf{H}^\top\mathbf{X}\mathbf{X}^\top\mathbf{H} \\ &\quad + 2\mathbf{X}\mathbf{X}^\top\mathbf{H}\mathbf{H}^\top\mathbf{H} + 2\lambda\mathbf{D}\mathbf{H} + \Phi \end{aligned} \quad (12)$$

Then, we multiply (12) element wise by H_{ij}^4 :

$$\begin{aligned} 4[\mathbf{X}\mathbf{X}^\top\mathbf{H}]_{ij}H_{ij}^4 + 2\lambda[\mathbf{S}\mathbf{H}]_{ij}H_{ij}^4 &= 2[\mathbf{H}\mathbf{H}^\top\mathbf{X}\mathbf{X}^\top\mathbf{H}]_{ij}H_{ij}^4 \\ &\quad + 2[\mathbf{X}\mathbf{X}^\top\mathbf{H}\mathbf{H}^\top\mathbf{H}]_{ij}H_{ij}^4 + 2\lambda[\mathbf{D}\mathbf{H}]_{ij}H_{ij}^4 \\ &\quad + \Phi_{ij}H_{ij}^4 \end{aligned}$$

Finally, using the KKT conditions $\Phi_{ij}H_{ij} = 0$, re-arranging and replacing back $\mathbf{X} \leftarrow \mathbf{X}^\top$ yields the update rule (8). Next, we focus on the convergence of the update rule where we use the auxiliary function technique.

Definition: $F(\mathbf{H}^{t+1}, \mathbf{H}^t)$ is an auxiliary function of $J(\mathbf{H}^t)$ if the following two conditions hold:

$$F(\mathbf{H}^{t+1}, \mathbf{H}^t) \geq J(\mathbf{H}^t) \text{ and } F(\mathbf{H}^t, \mathbf{H}^t) = J(\mathbf{H}^t)$$

Lemma: if F is an auxiliary function then $J(\mathbf{H})$ is non-increasing under the update rule: $\mathbf{H}^{t+1} = \arg \min_{\mathbf{H}} F(\mathbf{H}^{t+1}, \mathbf{H}^t)$.

We can show that:

$$\begin{aligned} F(\mathbf{H}^{t+1}, \mathbf{H}^t) &= \|\mathbf{X}\|^2 \\ &\quad - 2 \sum_{ijk} (H_{ji}^t(\mathbf{X}\mathbf{X}^\top)_{jk}H_{ki}^t(1 + \log \frac{H_{ji}^{t+1}H_{ki}^{t+1}}{H_{ji}^tH_{ki}^t})) \\ &\quad + \frac{1}{2} \sum_{ji} (\mathbf{H}^t\mathbf{H}^{t\top}\mathbf{X}\mathbf{X}^\top\mathbf{H}^t + \mathbf{X}\mathbf{X}^\top\mathbf{H}^t\mathbf{H}^{t\top}\mathbf{H}^t)_{ji} \frac{H_{ji}^{4,t+1}}{H_{ji}^{3,t}} \end{aligned}$$

is an auxiliary function for J_{data} , where $H_{ji}^{x,t}$ denotes the element (j,i) of the matrix \mathbf{H} raised to the power of x at the t iteration. The minimum value is obtained by setting $\frac{\partial F(\mathbf{H}^{t+1}, \mathbf{H}^t)}{\partial H_{jk}^{t+1}} = 0$ which yields (8) when no spatial term is used, i.e. $\lambda = 0$. When the spatial term is added, we can use the same procedure by taking into account the additional $\lambda\text{Tr}(\mathbf{H}^\top\mathbf{L}\mathbf{H})$ term. Then, we use the same auxiliary function $F(\mathbf{H}^{t+1}, \mathbf{H}^t)$ by adding the following term:

$$\lambda \sum_{i=1}^n \sum_{k=1}^c \frac{(\mathbf{L}\mathbf{H}^t)_{ik}H_{ik}^{2,t+1}}{H_{ik}^t} \quad (13)$$

Finally, we derive the correctness and convergence of (8) $\forall \lambda > 0$.

VII. REFERENCES

- [1] X. Pei, T. Wu, and C. Chen, "Automated graph regularized projective nonnegative matrix factorization for document clustering," *IEEE Trans. Cybernetics*, vol. 44, no. 10, pp. 1821–1831, Oct. 2014.
- [2] C. Ding, T. Li, and M. I. Jordan, "Convex and semi-nonnegative matrix factorizations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 1, pp. 45–55, Jan. 2010.
- [3] Z. Yuan and E. Oja, "Projective nonnegative matrix factorization for image compression and feature extraction," in *Proc. of 14th Scandinavian Conf. Image Anal.*, 2005, pp. 333–342.
- [4] Z. Tian and R. Kuang, "Global linear neighborhoods for efficient label propagation," in *Proc. of the SIAM Int'l Conf. on Data Mining (SDM)*, 2012, pp. 863–872.
- [5] C. Ding, T. Li, and W. Peng, "Nonnegative Matrix Factorization and Probabilistic Latent Semantic Indexing: Equivalence Chi-square Statistic, and a Hybrid Method," in *Proc. Nat'l Conf. on Artif. Intell. (AAAI)*, vol. 21, no. 1, 2006, p. 342.
- [6] C. Ding, X. He, and H. D. Simon, "On the equivalence of nonnegative matrix factorization and spectral clustering," in *Proc. of the SIAM Int'l Conf. on Data Mining (SDM)*, vol. 5, 2005, pp. 606–610.
- [7] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1548–1560, 2011.
- [8] Z. Yang, T. Hao, O. Dikmen, X. Chen, and E. Oja, "Clustering by nonnegative matrix factorization using graph random walk," in *Proc. Advances in Neural Information Processing Systems (NIPS)*, 2012, pp. 1079–1087.
- [9] J. Wen, Z. Tian, X. Liu, and W. Lin, "Neighborhood preserving orthogonal PNMf feature extraction for hyperspectral image classification," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 6, no. 2, pp. 759–768, Apr. 2013.
- [10] S. Zafeiriou, A. Tefas, I. Buciu, and I. Pitas, "Exploiting discriminant information in nonnegative matrix factorization with application to frontal face verification," *IEEE Trans. Neural Netw.*, vol. 17, no. 3, pp. 683–695, May 2006.
- [11] N. Guan, X. Zhang, Z. Luo, D. Tao, and X. Yang, "Discriminant projective non-negative matrix factorization," *PLoS ONE*, 2013.
- [12] X. Zhang, N. Guan, Z. Jia, X. Qiu, and Z. Luo, "Semi-supervised projective non-negative matrix factorization for cancer classification," *PloS one*, vol. 10, p. e0138814, 2015.
- [13] H. Kim and H. Park, "Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis," *Bioinformatics*, vol. 23, no. 12, pp. 1495–1502, 2007.
- [14] M. M. Kalayeh, H. Idrees, and M. Shah, "NMF-KNN: image annotation using weighted multi-view non-negative matrix factorization," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2014, pp. 184–191.
- [15] N. Guan, D. Tao, Z. Luo, and J. Shawe-Taylor, "MahNMF: Manhattan non-negative matrix factorization," *arXiv preprint arXiv:1207.3438*, 2012.
- [16] J. Yuan, D. Wang, and A. M. Cheryadat, "Factorization-based texture segmentation," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3488–3497, Nov. 2015.
- [17] D. D. Lee and S. H. Sebastian, "Algorithms for non-negative matrix factorization," in *Proc. Advances in Neural Information Processing Systems (NIPS)*, 2001, pp. 556–562.
- [18] Z. Yang and E. Oja, "Linear and nonlinear projective nonnegative matrix factorization," *IEEE Trans. Neural Netw.*, vol. 21, no. 5, pp. 734–749, 2010.
- [19] C. Boutsidis and E. Gallopoulos, "SVD based initialization: A head start for nonnegative matrix factorization," *Pattern Recognition*, vol. 41, no. 4, pp. 1350–1362, 2008.
- [20] X. Liu, S. Yan, and H. Jin, "Projective nonnegative graph embedding," *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1126–1137, May 2010.
- [21] Z. Yang and E. Oja, "Unified development of multiplicative algorithms for linear and quadratic nonnegative matrix factorization," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 1878–1891, Dec. 2011.
- [22] H. Zhang, Z. Yang, and E. Oja, "Adaptive multiplicative updates for quadratic nonnegative matrix factorization," *Neurocomputing*, vol. 134, pp. 206–213, 2014.
- [23] C. Bampis and P. Maragos, "Unifying the random walker algorithm and the SIR model for graph clustering and image segmentation," in *Proc. IEEE Int'l Conf. Image Processing (ICIP)*, Sept. 2015.
- [24] F. Meyer, "Topographic distance and watershed lines," *Signal processing*, vol. 38, no. 1, pp. 113–125, 1994.
- [25] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.
- [26] Y. He, H. Lu, and S. Xie, "Semi-supervised non-negative matrix factorization for image clustering with graph laplacian," *Multimedia tools and applications*, vol. 72, no. 2, pp. 1441–1463, 2014.
- [27] R. Rajabi and H. Ghassemian, "Hyperspectral data unmixing using GNMF method and sparseness constraint," in *Int'l Sympos. on Geos. and Rem. Sens.*, 2013, pp. 1450–1453.
- [28] D. Cai, X. He, X. Wang, H. Bao, and J. Han, "Locality preserving nonnegative matrix factorization," in *IJCAI*, vol. 9, 2009, pp. 1010–1015.
- [29] C. Ding, T. Li, W. Peng, and H. Park, "Orthogonal nonnegative matrix t-factorizations for clustering," in *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006, pp. 126–135.
- [30] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using gabor filters," *Pattern Recognition*, vol. 24, pp. 1167–1186, 1991.