Hessian non-negative hypergraph

Abstract. Hypergraph is a vital relationship modeling tool applied to many learning tasks. Compared with the common graph, the hypergraph can model the high-order relationship among more than two data samples. Thus, hypergraph provides more flexibility and the capability of modeling complex data correlations. However, real-world image data inevitably lie on or close to a thin layer of a manifold while containing the noise component. The practical image data influence the hypergraph learning procedure, subsequently resulting in a notable performance deterioration. To this end, we propose a novel Hessian non-negative hypergraph model. Specifically, the Hessian energy regularized non-negative constrained data reconstruction is used to generate manifold-respecting and noise resistant hyperedges. The extensive experiments on the benchmark image datasets demonstrate that the proposed hypergraph model can significantly enhance the learning performance of the hypergraph-based image clustering and classification.

Keywords: Hypergraph; Hessian energy; Non-negative; Data representation.

1 Introduction

The development of various data acquisition technologies, such as digital cameras, has resulted in a very fast accumulation of high-dimensional visual data. Effective data analysis algorithms are designed to extract important and useful information to enhance life quality and spawn new business models. Graph-based learning models [1] have received considerable attention among various algorithms. Hypergraph is used as the extension of graphs, which can model the high-order relationship among the data samples. Thus, hypergraph-based learning methods have achieved promising performance on several visual applications, such as topic-sensitive influencer mining [2], image segmentation [3,4], image clustering [5-7] and image classification [8,9].

For the hypergraph-based learning tasks, an informative hypergraph that can effectively modulate the underlying manifold of the data must be constructed. A neighborhood-based strategy [2,4] is commonly used to generate a hyperedge set, where each

[*] Corresponding Author: liujxxxy@126.com (Jie Liu)

hyperedge provides the capability to model complex correlations of data. In detail, this strategy takes each sample as a central node and then selects the k-nearest-neighbors to generate each hyperedge. The entire hypergraph is constructed based on a hyperedge set. However, the neighborhood-based strategy is sensitive to the neighborhood size and is only suitable for uniform distribution data. Moreover, noise usually contaminates the real-world data and the noisy data samples may be linked to generate a set of hyperedges. Thus, the learning performance of the neighborhood-based models is degraded dramatically.

To tackle the noise issue, some recent studies have attempted to use data representation to generate noise-resistant hyperedges. For instance, ℓ_1- Hypergraph [8] leverages sparse representation to reconstruct each sample and select the samples with sparse codes to generate each hyperedge. Furthermore, ℓ_1-Hypergraph is extended by varying the regularization values of sparse representation and modulating different weighting schemes of hyperedges [9]. ℓ_2-Hypergraph [5] leverages ridge regression instead of sparse coding for hyperedge generation. To be robust to significant data errors such as sample-specific outliers, Elastic-net Hypergraph [7] generates flexible hyperedges by solving a matrix elastic-net problem. As only the data samples with large representation coefficients are linked for hyperedge generation, the representation-based hypergraph models, to some extent, are not sensitive to noise.

However, the existing representation-based hypergraph construction methods cannot still sufficiently reflect the data manifold [5-7]. Specifically, real-world image data lie on or close to a thin layer of a manifold embedded into high dimensional space. At the same time, the sampled image data may be contaminated by noise. For such image data, the existing hypergraph construction models including the neighborhood-based and representation-based methods cannot effectively capture the high-order correlations of the image data, resulting in degenerating learning performance.

To this end, we aim to solve the practical problem of hypergraph learning with noisy image data sampled from the underlying manifold. The contributions of this article are summarized as follows:

1) We propose a novel hypergraph learning method, which introduces Hessian energy to the process of hyperedge generation. Hessian energy can make data representation vary linearly over the geodesics of the underlying image manifold.

2) The proposed method generates a set of hyperedges based on Hessian energy regularized data representation. Specifically, the non-negative constraint is imposed on the data representation. Then, the resultant large data representation is used to choose the image samples for hyperedge generation.

3) We employ extensive experiments on the noisy image datasets, demonstrating superior learning performance of the proposed method over the baseline methods.

2 Related works

Hypergraph has been used to model complex relationship among the data samples for the clustering task. For instance, Fang et al. [2] proposed a topic-sensitive influence mining framework and used hypergraph to determine the influence estimation. Docournau et al. [3] exploited directed hypergraphs to solve an image segmentation problem. Furthermore, Kim et al. [4] suggested a hypergraph-based image segmentation method, employing a higher-order correlation clustering for a hypergraph-based segmentation procedure. Jin et al. [5] proposed ℓ_2-Hypergraph, where the hyperedges are generated by solving a ridge regression problem. To be robust to non-Gaussian noises, Jin et al. [6] put forth a correntropy-induced-Hypergraph model, which leverages the correntropy-induced metric to measure the data errors and imposes a low-rank constraint on data representation. Liu et al. [7] proposed the Elastic-net Hypergraph, which generates a set of hyperedges by solving a matrix elastic-net problem.

Except for the clustering task, the hypergraph is also used for the image classification and image ranking tasks. For instances, Zhang et al. [12] revisited semi-supervised learning on hypergraphs, confirming the interval approach's efficacy on hypergraphs. Yu et al. [11] formulated a hypergraph-based semi-supervised learning method, where the weighting of hyperedges is adaptively coordinated. An et al. [13] proposed a multi-hypergraph-based person re-identification method, which leverages multiple hypergraphs to fuse complementary information embedded in different feature representations.

Hypergraph is used for other learning tasks. For instance, Ji et al. [14] used a bi-layer multimodal hypergraph learning method for the robust sentiment prediction of multi-modal microblog tweets. Jin et al. [10] incorporated the neighborhood hypergraph to the low-rank matrix factorization framework, which makes the derived data representation capture the underlying manifold. Zhang et al. [9] used a hypergraph learning approach for feature selection, simultaneously learning hyperedge weights and doing feature selection.

All these methods have demonstrated that hypergraph can significantly enhance the performance of machine learning approaches when capturing the high-order relationship among the data samples. Thus, this article aims to introduce the Hessian energy operator to hypergraph learning, which results in a manifold-respecting hypergraph.

3 Methodology

Given a data matrix, $\mathbf{X} = [\mathrm{x}_1, \mathrm{x}_2, \cdots, \mathrm{x}_m] \in \mathbb{R}^{n \times m}$, each column $x_i \in \mathbb{R}^n$ is image data vector, which corresponds to a node of a hypergraph. To model the high-order image correlations, each hyperedge of a hypergraph links the different numbers of images. Thus, hypergraph can be represented as $HG = (V, E, \boldsymbol{W})$, where V is a set of images known as nodes or vertices, and E is a set of non-empty subsets of V as hyperedges and a hyperedge e usually contains more than two nodes. A hyperedge weights matrix $\boldsymbol{W}$ is defined to model the different effectiveness of the hyperedges.

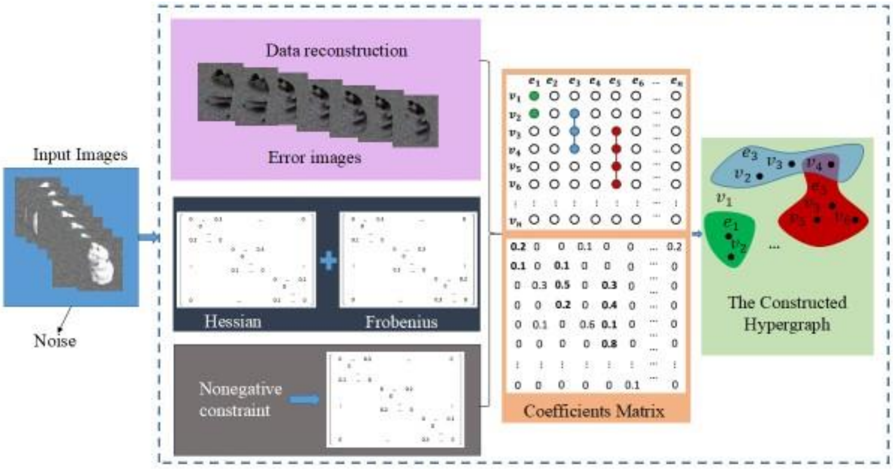


Fig. 1. The framework of the proposed method. For the noisy image data (the images contaminated by Gaussian noise), the images are reconstructed linearly and the Frobenius norm is used to measure the reconstruction errors. Both the Hessian energy and Frobenius-norm are used to regulate the data representation. Finally, the non-negative constraint is imposed on the data representation to derive the parts-based data representation. Furthermore, the images with large data representation are used to generate a manifold respecting and noise-resistant hyperedges.

3.1 Hessian energy-based manifold preserving

Considering that $f(\mathrm{x}_i)$ is a function that maps the image x_i to the corresponding data representation, we define $\|f_k\|_M^2$ to measure the smoothness of $f_k(\boldsymbol{x}_i)$ over the geodesics on image manifold. Hessian energy [21] is an important tool to measure the smoothness of a function, defined as

$$\|f_k\|_M^2 = \int_M \|\nabla_a \nabla_b f_k\|^2_{T_x^*M \otimes T_x^*M} dV(\boldsymbol{x}) \tag{1}$$

where $\nabla_a \nabla_b f$ is the second covariant derivative of f. In addition, $dV(\boldsymbol{x})$ is the natural volume element. $\|\nabla_a \nabla_b f\|^2$ is the ℓ_2-norm of the second covariant derivative, calculated as the Frobenius-norm of the Hessian of f in normal coordinates.

Given normal coordinates x_r at x_i and the function values $f(\mathrm{x}_j)$ on $N_k(\mathrm{x}_i)$, we would like to have an operator H to estimate the Hessian of f at x_i, *i.e.*,

$$\left.\frac{\partial^2 f_k}{\partial x_r \partial x_s}\right|_{X_i} \approx \sum_{j=1}^{p} \boldsymbol{H}_{rsj}^{(i)} f_k(\boldsymbol{x}_j) \tag{2}$$

Furthermore, we can calculate the Hessian energy operator **H** via a secondorder polynomial curving fitting in normal coordinates, *i.e.*,

$$q(x_i) = f_k(\boldsymbol{x}_j) + \sum_{r=1}^{n} B_r \boldsymbol{x}_r + \sum_{r}^{n} \sum_{s=r}^{n} A_{rs} \boldsymbol{x}_r \boldsymbol{x}_s \tag{3}$$

where $q(x_i)$ is the second-order polynomial and the zeroth-order term is fixed at $f(\mathrm{x}_i)$. If the neighborhood size is zero, it becomes the second-order Taylor expansion of f around x_i. *i.e.*, $B_r = \frac{\partial f_k}{\partial x^r}\Big|_{x_i}$ and $A_{rs} = \frac{\partial^2 f_k}{\partial x^r \partial x^s}\Big|_{x_i}$.

To estimate the local tangent space, we measure the smoothness of the function on the data locality. In other words, we perform PCA on the k-nearest neighbors of each image, and the derived t leading eigenvectors correspond to an orthogonal basis of the local tangent space. Furthermore, we leverage linear least squares to fit the second-order polynomial, thus resulting in

$$arg\min_{w \in R^S} \sum_{j=1}^{k} \left(f_k(\boldsymbol{x}_j) - f_k(\boldsymbol{x}_i) - (\boldsymbol{\Phi w})_j\right)^2 \tag{4}$$

where $\Phi \in \mathbb{R}^{k \times P}$ is the design matrix with $P = m + \frac{m(m+1)}{2}$. The last $\frac{m(m+1)}{2}$ components of w corresponding to the coefficients $A_{rs}$of polynomial.

Based on Eq. (4), we can derive the desired form $\mathbf{H}_{rsj}^{(i)}$. The estimation of the Frobenius-norm of the Hessian energy of f at X_i is formulated as

$$||\nabla_a \nabla_b f_k||^2 \approx \sum_{r,s=1}^{M} \left(\sum_{\alpha=1}^{p} H_{rs\alpha}^{(i)}\right)^2 = \sum_{\alpha,\beta=1}^{p} v_{\alpha k} v_{\beta k}\, \mathbf{L}_{\alpha\beta}^{(i)} \tag{5}$$

where $\mathbf{L}_{\alpha\beta}^{(i)} = \sum_{r,s=1}^{n} \mathbf{H}_{rs\alpha}^{(i)} \mathbf{H}_{rs\beta}^{(i)}$. Then, the estimated Hessian energy R_p is the sum over all the data points, defined as

$$R_k = \sum_{i=1}^{m} \sum_{r,s=1}^{n} \left(\frac{\partial^2 f_k}{\partial x_r \, \partial x_j}\Big|_{x_i}\right)^2 = \sum_{i=1}^{m} \sum_{\alpha \in N_p(X_i)} \sum_{\beta \in N_p(X_i)} v_{\alpha k} v_{\beta k} \mathbf{L}_{\alpha\beta}^{(i)} \tag{6}$$

where m is the number of data points.

3.2 Hessian regularized non-negative data representation.

Based on the self-expression property of the image data, we reconstruct each image by the entire dataset. In other words, for each image x_i, it can be represented as a new coordinate vector with respect to the entire image dataset. Furthermore, we combine the Hessian energy regularization term into the data reconstruction framework, resulting in the following optimization problem:

$$\begin{aligned} &\min_{\mathbf{F}} ||\mathbf{X} - \mathbf{XF}||_F^2 + \lambda \sum_k R_k + \gamma ||\mathbf{F}||_F^2 \\ &= \min_{\mathbf{F}} ||\mathbf{X} - \mathbf{XF}||_F^2 + \lambda \mathrm{Tr}(\mathbf{F} \mathbf{L}_{Hess} \mathbf{F}^T) + \gamma ||\mathbf{F}||_F^2 \\ &\text{s.t. } \mathbf{F} \geq 0 \end{aligned} \tag{7}$$

where $Tr(\cdot)$ denotes trace of the matrix. $\mathbf{L}_{Hess} = \sum_{i=1}^{m} \mathbf{L}^{(i)}$.

3.3 Hyperedge generation

Given a node set and a hyperedge set, an incidence matrix $\mathbf{H}$ indicates whether a node belongs to a hyperedge. In other words, $\mathbf{H}(v, e) = 1$ if $v \in e$; otherwise, $\mathbf{H}(v, e) = 0$.

The hyperedges have flexibility to model the high-order image correlations, which links each image to different numbers of images. We need compute the incidence matrix to indicate the relationship between each image and a hyperedge. Thus, we compute the incidence matrix as

$$\mathbf{H}(v_j, e_i) = \begin{cases} 1 & \text{if } f_{ij} \in top(q) \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

where e_i is the hyperedge associated with data point $\mathbf{x}_i$, and f_{ij} is the j-th representation coefficient of $\mathrm{x}_i$and $top(q)$ is the top-q-largest data representation of x_i.

Finally, we present the weight scheme for a given hyperedge. In our method, we sum up the similarities of the images as the weight of a hyperedge, where the similarity between two images is calculated as the dot product of the two representation vectors of the two images. Thus, we give the weight of the hyperedge as [6]

$$w(e_i) = \sum_{v_i \in e_i, j \neq i} S(\mathbf{f}_i, \mathbf{f}_j) \quad (9)$$

where $S(\mathbf{f}_i, \mathbf{f}_j) =< \mathbf{f}_i, \mathbf{f}_j >$ is used to measure the similarity between x_i and x_j.

3.4 The proposed algorithm

Algorithm 1 lists the main procedure of how to construct a Hessian-induced non-negative hypergraph.

Algorithm 1. Hessian non-negative hypergraph

```
Input: Image data matrix $\mathbf{X}$
Output: Hessian hypergraph
Input the image data matrix
Solve the optimization problem of Eq. (7)
for $i$ = 1, · · ·, $m$ do
    Define the incidence vector according to Eq. (8).
    Link the images with large data representation to $\mathrm{x}_i$.
    Calculate the weight of hyperedge $e_i$ according to Eq. (9).
end for
Output a Hessian non-negative hypergraph.
```

4 Optimization

4.1 The update rule

Notice that $||\mathbf{A}||_F^2 = \mathrm{Tr}(\mathbf{A}\mathbf{A}^T)$, we have

$$\begin{aligned} O &= \mathrm{Tr}((\mathbf{X}-\mathbf{XF})(\mathbf{X}-\mathbf{XF})^T) + \lambda \mathrm{Tr}(\mathbf{F}\mathbf{L}_{Hess}\mathbf{F}^T) + \gamma \mathrm{Tr}(\mathbf{F}\mathbf{F}^T) \\ &= \mathrm{Tr}(\mathbf{X}\mathbf{X}^T - 2\mathbf{XF}\mathbf{X}^T + \mathbf{XF}\mathbf{F}^T\mathbf{X}^T + \lambda \mathbf{F}\mathbf{L}_{Hess}\mathbf{F}^T + \gamma \mathbf{F}\mathbf{F}^T) \end{aligned} \quad (10)$$

Let ϕ_{kj} be the Lagrange multiplier for constraint $f_{kj} \geq 0$. We define matrix $\mathbf{\Phi} = [\phi_{kj}]$, then the Lagrange function Q is

$$Q = \mathrm{Tr}\big(\mathbf{X}\mathbf{X}^T - 2\mathbf{XF}\mathbf{X}^T + \mathbf{XF}\mathbf{F}^T\mathbf{X}^T + \lambda \mathbf{F}\mathbf{L}_{Hess}\mathbf{F}^T + \gamma \mathrm{Tr}(\mathbf{F}\mathbf{F}^T)\big) + \mathrm{Tr}(\mathbf{\Phi}\mathbf{F}^T) \quad (11)$$

The partial derivatives of Q with respect to $\mathbf{F}$ is

$$\frac{\partial Q}{\partial \mathbf{F}} = -2\mathbf{X}^T\mathbf{X} + 2\mathbf{X}^T\mathbf{X}\mathbf{F} + 2\lambda\mathbf{F}\mathbf{L}_{Hess} + \gamma\mathbf{F} + \mathbf{\Phi} \tag{12}$$

Using the KKT condition $\phi_{kj}f_{kj} = 0$, we get the following formulation for f_{kj}:

$$(\mathbf{X}^T\mathbf{X}\mathbf{F})_{kj}f_{kj} + (\lambda\mathbf{F}\mathbf{L}_{Hess})_{kj}f_{kj} = (\mathbf{X}^T\mathbf{X})_{kj}f_{kj} + \gamma f_{kj} \tag{13}$$

We adopt a similar trick as in [17] to solve the optimization problem. Introduce

$$\mathbf{L}_{Hess} = \mathbf{L}_{Hess}^{+} - \mathbf{L}_{Hess}^{-} \tag{14}$$

where $(\mathbf{L}_{Hess}^{+})_{ij} = \frac{|(\mathbf{L}_{Hess})_{ij}|+(\mathbf{L}_{Hess})_{ij}}{2}$ and $(\mathbf{L}_{Hess}^{-})_{ij} = \frac{|(\mathbf{L}_{Hess})_{ij}|-(\mathbf{L}_{Hess})_{ij}}{2}$.

Substituting Eq. (14) into Eq. (13), we derive the following formulation:

$$-(\mathbf{X}^T\mathbf{X})_{kj}\mathbf{F}_{kj} - (\lambda\mathbf{F}\mathbf{L}_{Hess}^{-})_{kj}\mathbf{F}_{kj} + (\mathbf{X}^T\mathbf{X}\mathbf{F})_{kj}\mathbf{F}_{kj} + (\lambda\mathbf{X}\mathbf{L}_{Hess}^{+})_{kj}\mathbf{F}_{kj} + \gamma\mathbf{F}_{kj} = 0 \tag{15}$$

Eq. (15) results in the following update rules:

$$\mathbf{F}_{kj} \leftarrow \mathbf{F}_{kj}\frac{(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{F}\mathbf{L}_{Hess}^{-})_{kj}}{(\mathbf{X}^T\mathbf{X}\mathbf{F} + \lambda\mathbf{F}\mathbf{L}_{Hess}^{+} + \gamma\mathbf{F})_{kj}} \tag{16}$$

4.2 Convergence Analysis

The update rule of Eq. (16) is literately employed to update $\boldsymbol{F}$. In the following, we give the proof to prove the objective function is non-increasing under this update rule.

Definition:$G(f, f')$ is an auxiliary function for $g(f)$ if the conditions

$$G(f, f') \geq g(f), G(f, f) = g(f) \tag{17}$$

are satisfied.

Lemma: If G is an auxiliary function of g, then g is non-increasing under the update

$$f^{(t+1)} = arg\min_{f} G\left(f, f^{(t)}\right) \tag{18}$$

Proof:

$$g\left(f^{(t+1)}\right) \leq G\left(f^{(t+1)}, f^{(t)}\right) \leq G\left(f^{(t)}, f^{(t)}\right) = g\left(f^{(t)}\right) \tag{19}$$

For each element f_{kj} in $\mathbf{F}$, g_{kj} denotes the part of the objective function which is relevant only to f_{kj}.

$$g'_{kj} = \left(\frac{\partial O}{\partial \mathbf{F}}\right)_{kj} = (-2\mathbf{X}^T\mathbf{X} + 2\mathbf{X}^T\mathbf{X}\mathbf{F} + 2\lambda\mathbf{F}\mathbf{L}_{Hess} + 2\gamma\mathbf{F})_{kj} \tag{20}$$

$$g''_{kj} = 2(\mathbf{X}^T\mathbf{X})_{kk} + 2\lambda\mathbf{F}_{kj}^{(t)}(\mathbf{L}_{Hess})_{jj} + 2\gamma\mathbf{F} \tag{21}$$

Lemma: Function

$$\begin{aligned} G\left(f, f_{kj}^{(t)}\right) &= g_{kj}\left(f_{kj}^{(t)}\right) + g'_{kj}\left(f_{kj}^{(t)}\right)\left(f - f_{kj}^{(t)}\right) \\ &+ \frac{(\mathbf{X}^T\mathbf{X}\mathbf{F} + \lambda\mathbf{F}\mathbf{L}_{Hess}^{+} + \gamma\mathbf{F})_{kj}}{f_{kj}^{(t)}}(f - f_{kj}^{(t)})^2 \end{aligned} \tag{22}$$

is an auxiliary function for g_{kj}, the part of the objective function which is relevant only to f_{kj}.

Proof: First, we obtain $G(f, f) = g_{kj}(f)$, so we need to prove only that $G(f, f_{kj}^{(t)}) \geq g_{kj}(f)$. The Tayler series expansion of $g_{kj}(f)$ is as follows:

$$\begin{aligned} g_{kj}(f) &= g_{kj}\left(f_{kj}^{(t)}\right) + g'_{kj}\left(f_{kj}^{(t)}\right)\left(f - f_{kj}^{(t)}\right) \\ &+ [(\mathbf{X}^T\mathbf{X})_{kk} + \lambda(\mathbf{L}_{Hess})_{jj} + +\gamma\mathbf{F}](f - f_{kj}^{(t)})^2 \end{aligned} \tag{23}$$

We have

$$(\mathbf{X}^T\mathbf{XF})_{kj} = \sum_{l=1}^{K} (\mathbf{X}^T\mathbf{X})_{kl} f_{lj}^{(t)} \geq f_{kj}^{(t)} (\mathbf{X}^T\mathbf{X})_{kk} \tag{24}$$

and

$$(\mathbf{FL}_{Hess}^{+})_{kj} = \sum_{l=1}^{N} f_{kl}^{(t)} (\mathbf{L}_{Hess}^{+})_{lj} \geq \mathbf{f}_{kj}^{(t)} (\mathbf{L}_{Hess}^{+})_{jj}$$

$$\geq f_{kj}^{(t)} (\mathbf{L}_{Hess}^{+} - \mathbf{L}_{Hess}^{-})_{jj} = f_{kj}^{(t)} (\mathbf{L}_{Hess})_{jj} \tag{25}$$

Then, we arrive at: $\frac{(\mathbf{X}^T\mathbf{XF}+\lambda\mathbf{VL}_{Hess}^{+})_{kj}}{f_{kj}^{(t)}} \geq (\mathbf{X}^T\mathbf{X})_{kk} + \lambda(\mathbf{L}_{Hess})_{jj}$. Thus, $G(f, f_{kj}^{(t)}) \geq g_{kj}(f)$.

Lemma: The objective function in Eq. (10) is non-increasing under the updating rules in Eq. (16).

Proof: Replacing $G(v, v_{kj}^{(t)})$ in Eq. (18) by Eq. (22) results in the update rule as:

$$f_{kj}^{(t+1)} = f_{kj}^{(t)} - f_{kj}^{(t)} \frac{g'_{kj}\left(f_{kj}^{(t)}\right)}{(2\mathbf{X}^T\mathbf{XF} + 2\lambda\mathbf{FL}_{\mathrm{Hess}}^{+} + 2\gamma\mathbf{F})_{kj}}$$

$$= f_{kj}^{(t)} \frac{(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{FL}_{Hess}^{-})_{kj}}{(\mathbf{X}^T\mathbf{XF} + \lambda\mathbf{FL}_{Hess}^{+} + \gamma\mathbf{F})_{kj}} \tag{26}$$

Because Eq. (22) is an auxiliary function, g_{kj} is non-increasing under this update rule.

5 Experimental results and discussing

5.1 Datasets and baselines

We validate the proposed method on two real-world image datasets Coil 100 [22] and USPS [5]. To demonstrate the superiority of our method, we compare it with multiple baselines including the neighborhood-based common graph (KNN-Gr), Ada-Hypergraph (Ada-HG) [11], KNN-Hypergraph (KNN-HG) [16], ℓ_1-Hypergraph (ℓ_1-HG) [8], ℓ_2-Hypergraph (ℓ_2-HG) [5] and Elastic-Net Hypergraph (EN-HG) [7]. We conduct cross-validation tests to tune the parameter values of the baseline methods.

5.2 Image clustering on corrupted data

Hypergraph-based clustering belong to the graph-based machine learning model. When the constructed hypergraph is used to model the relationship among the images, the corresponding hypergraph Laplacian is embedded into the spectral clustering framework to employ the image clustering task. To simulate the noisy image data, we add the different corruptions of Gaussian noise to the image x of each dataset. In other words, $\mathrm{x} = \mathrm{x} + \eta\mathrm{n}$, where η is the corruption ratio and n is the noise following the Gaussian distribution. To measure the image clustering performance of different methods, we adopt classification accuracy (AC) and the mutual information-based metric (NMI) [10] to evaluate the performance. The results of different methods are listed in Table 1 and Table 2. From two tables, we observe the following:

Table 1. The image clustering results on Coil 100 dataset

Method	AC(%)				NMI(%)			
Corrupted ratio	10%	20%	30%	40%	10%	20%	30%	40%
KNN-Gr	70.9	65.4	61.2	56.3	79.5	74.3	69.7	63.2
KNN-HG [16]	71.9	67.5	63.2	57.6	80.2	76.3	72.5	67.2
EN-HG [7]	70.1	68.5	66.3	64.2	80.2	77.5	75.6	73.2
Ada-HG [11]	73.2	70.6	64.3	60.1	86.3	80.3	75.2	69.5
ℓ_1-HG [8]	72.5	70.1	68.9	66.5	83.2	81.5	79.5	77.6
ℓ_2-HG [5]	74.3	72.1	70.3	68.5	85.2	83.5	81.1	77.5
Hs-HG (ours)	75.6	73.1	71.2	69.8	86.5	84.1	82.1	79.6

Table 2. The image clustering results on USPS 100 dataset

Method	AC(%)				NMI(%)			
Corrupted ratio	10%	20%	30%	40%	10%	20%	30%	40%
KNN-Gr	68.5	64.3	60.3	56.2	72.3	66.5	62.9	56.8
KNN-HG [16]	70.2	66.3	61.5	57.6	73.6	69.2	64.2	60.3
EN-HG [7]	74.3	73.1	71.2	69.1	76.5	73.5	70.2	68.5
Ada-HG [11]	78.9	74.1	70.2	64.3	80.3	76.2	71.2	64.2
ℓ_1-HG [8]	76.5	74.8	75.2	72.8	78.9	75.4	73.1	72.3
ℓ_2-HG [5]	80.2	78.5	76.2	74.3	81.5	80.0	78.5	75.3
Hs-HG (ours)	81.3	79.6	77.2	76.3	82.9	81.2	79.6	77.5

The proposed hypergraph-based image clustering method can outperform the second-best results of the baseline method by at least 1% and 2% on the Coil 100 and USPS image datasets, respectively. In particular, the performance gain of the proposed method (Hs-HG) is more evident as the image noise corruption level increases. The reason is that the Hessian energy is introduced to the data reconstruction scheme, which can be robust to image noise and discover the discriminative structure of the underlying image manifold. Moreover, the non-negative constraint makes the data reconstruction scheme learn the parts-based data representation. Thus, Hessian energy, data reconstruction scheme and parts-based data representation make the proposed method achieve superior performance over the baseline methods.

Table 3. The image classification results on Coil 100 dataset

Method	ACC(%)			
Corrupted ratio	10%	20%	30%	40%
KNN-Gr	89.5	84.3	79.2	75.6
KNN-HG [16]	90.6	86.5	81.3	77.9
EN-HG [7]	88.6	86.2	84.1	82.2
Ada-HG [11]	91.2	88.3	83.1	79.2
ℓ1-HG [8]	91.2	89.2	86.3	83.5
ℓ2-HG [5]	92.1	90.2	88.4	86.5
Hs-HG(ours)	94.1	93.1	91.2	89.1

Table 4. The image classification results on USPS dataset

Method	ACC(%)			
Corrupted ratio	10%	20%	30%	40%
KNN-Gr	94.5	91.2	85.3	80.5
KNN-HG [16]	96.1	92.3	88.7	83.2
EN-HG [7]	93.1	90.4	89.8	87.6
Ada-HG [11]	95.4	93.4	89.2	83.1
ℓ1-HG [8]	95.6	93.2	91.1	89.8
ℓ2-HG [5]	96.2	94.5	90.2	91.0
Hs-HG(ours)	97.2	95.8	94.2	92.2

5.3 Image classification on corrupted data

We use the common transductive classification framework [5,8] to classify. We adopt the classification accuracy for classification (ACC) [6] to evaluate the classification performance and leverage the same noisy image datasets, which is the same as the ones used in the clustering experiments. For Coil 100 dataset, we randomly select 30% of the labeled images from each object to form the training set and then the rest of the object images as the test set. For USPS dataset, we adopt the familiar experimental setting, which includes 7,291 train images and 2,007 test images. Table 3 and Table 4 list the experimental results on two corrupted datasets.

From two tables, we observe that the proposed method (Hs-HG) consistently outperforms the baseline methods and achieves the best image classification performance. The main reason is that the proposed method effectively captures the image manifold via Hessian energy regularization. Compared with the baseline methods, the hessian energy can make the data representation with the same class move more smoothly over the image manifold. Moreover, the non-negative constraint makes the data reconstruction scheme learn the parts-based data representation, resulting in more discriminative hyperedges.

Table 5. The ablation studies on Coil 100 dataset

Method	AC(%)			NMI(%)			ACC(%)		
Corrupted ratio	10%	20%	40%	10%	20%	40%	10%	20%	40%
He-HG	73.8	72.5	67.5	83.3	81.4	77.6	92.1	91.2	87.2
Hc-HG	70.4	68.5	64.3	76.5	73.2	71.5	88.3	86.2	80.1
Hs-HG	75.6	73.1	69.8	86.5	84.1	79.6	94.1	93.1	89.1

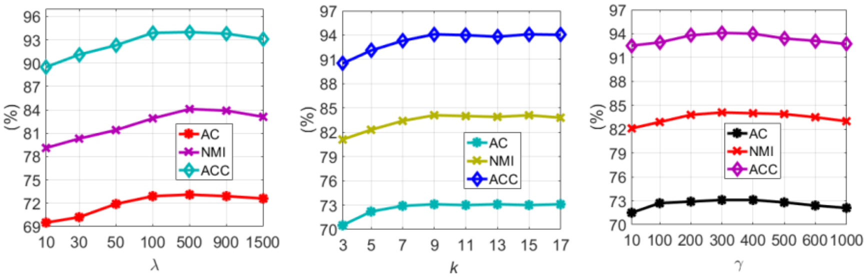
Table 6. The ablation studies on USPS dataset

Method	AC(%)			NMI(%)			ACC(%)		
Corrupted ratio	10%	20%	40%	10%	20%	40%	10%	20%	40%
He-HG	80.3	78.1	75.1	81.0	77.6	75.4	96.1	95.1	91.2
Hc-HG	78.6	77.5	72.0	78.9	77.1	71.8	92.3	90.4	87.3
Hs-HG	81.3	79.6	76.3	82.9	81.2	77.5	97.2	95.8	92.2

5.4 Ablation studies of two strategies

To validate further the effectiveness of the proposed two schemes, namely, the Hessian energy regularization and the non-negative constraint for representation-based hypergraph learning, we perform detailed ablation studies on two noisy image datasets to compare the proposed method with the following two variants. (1) We remove the Hessian energy regularization from the proposed framework and preserve the constraint, resulting in one variant of the proposed method, termed as Hc-HG. (2) We remove the non-negative constraint from the proposed framework and preserve the Hessian energy term, resulting in the other variant of the proposed method, termed as He-HG. The image clustering and classification results are listed in Table 5 and Table 6.

The two tables show that the model with Hessian energy regularization (He-HG) and the one with the non-negative constraint (Hc-HG) bring some performance improvements on two image noisy datasets.



(a) (b) (c)

Fig. 2. The performance under different parameters. (a) k, (b) λ and (c) γ.

With both strategies, the proposed method (Hs-HG) displays the large performance improvements over the model with the single strategy (Hc-HG or He-HG). The experimental results demonstrate that both the Hessian energy regularization and the non-negative constraint play the important roles in enhancing the performance of hypergraph-based learning tasks.

5.5 Parameter setting

The parameter adjustment is crucial for the proposed method to achieve encouraging results on different image datasets. We design the parameter setting for the proposed method, which has three essential regularization parameters: k, λ, and γ. k is the neighborhood size parameter, λ is the Hessian energy regularization parameter and γ is the Frobenius norm regularization parameter. For the q parameter, we preserve the top-5 most significant data representations and set the other data representations as zeros. We conduct the parameter setting experiments by varying different values of one parameter when fixing two parameters on the Coil 100 dataset (20% image Gaussian Noise). From Fig. 2, we observe the following:

With the increase of parameter k, the performance also increases accordingly. The learning performance of the proposed method begins to degenerate a little until the parameter value reaches 9.

When the parameter value of λ is too small or too large, the learning performance degenerates slightly. For the other parameter values, the learning performance is relatively stable.

The performance changes under different parameter values of γ are similar as the other parameters. In other words, when the parameter value of γ is very small, resulting in the degenerating performance; when the value of parameter γ is very large, the learning procedure is over-fitted to the regularization process.

6 Conclusion

In this article, we have proposed a novel Hessian-induced hypergraph model. Compared with the existing methods, the proposed model has two key differences: (1) Hessian energy-based data representation to generate the manifold respecting hyperedges. (2) The non-negative constraint is imposed on the data representation to generate noise-resistant hyperedges. The experimental results on the noisy image datasets demonstrate that the proposed hypergraph construction method outperforms the baselines with a large performance gain. In this work, we choose middle-size image datasets instead of large datasets to employ the proposed method. We will make other attempts to handle the scalability issue in the future.

Acknowledgements

This work is supported by National Key Research and Development Program of China (No.2020AAA0109700), by the National Natural Science Foundation of China (Nos. 62072386, 62076167) and by the Henan Center for Outstanding Overseas Scientists（GZS2022011）

References

1. M. Belkin, P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In Advances in Neural Information Processing Systems 14: 585-591. 2001.
2. Q. Fang, J. Sang, C. Xu,et al. Topic-sensitive influencer mining in interest-based social media networks via hypergraph learning. IEEE Transactions on Multimedia, 16(3): 796-812, 2014.
3. A. Ducournau, and A. Bretto. Random walks in directed hypergraphs and application to semi-supervised image segmentation. Computer Vision and Image Understanding, 2014,120:91-102
4. S. Kim, Chang D. Yoo, S.Nowozin, et al.Image Segmentation Using Higher Order Correlation Clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 36(9): 1761-1774, 2014.
5. T. Jin; Z. Yu,Y. Gao, S. Gao, X. Sun,C. Li. Robust L_2-Hypergraph and its applications. Information Sciences, 2019, 501: 708-723.
6. T. Jin, R. Ji, Y. Gao, X. Sun, X. Zhao, and D. Tao. Correntropy-induced robust low-rank hypergraph. IEEE Transactions on Image Processing, 2019, 28(6): 2755-2769

7. Q. Liu, Y. Sun, C. Wang, T. Liu, and D. Tao. Elastic-Net hypergraph learning for image clustering and semi-Supervised classification. IEEE Transactions on Image Processing, 2017, 26(1):452-463.
8. M. Wang, X. Liu, and X. Wu. Visual classification by ℓ1-hypergraph modeling. IEEE Transactions on Knowledge and Data Engineering, 2015, 27(9):2564-2574.
9. Z. Zhang, L. Bai, Y. Liang, and E. Hancockc. Joint hypergraph learning and sparse regression for feature selection. Pattern Recognition, 2017, 63:291- 309.
10. T. Jin, J. Yu,Y. Jane, et al. Low-rank matrix factorization with multiple hypergraph regularizer. Pattern Recognition. 48(3): 1011-1022,2015.
11. J. Yu, D. Tao, M. Wang. Adaptive Hypergraph learning and its application in image classification. IEEE Transactions on Image Processing, 21(7):3262-3272, 2012.
12. C. Zhang, S. Hu, Z. Tang, and T.H. Chan. Re-revisiting learning on hypergraphs: confidence interval, subgradient method and extension to multiclass. IEEE Transactions on Knowledge and Data Engineering, 2020,32(3):506-518.
13. L. An, X. Chen, and S. Yang, and X. Li. Person re-identification by multihypergraph fusion. IEEE Transactions on neural networks and learning systems, 2016, 28(11), 2763-2774.
14. R. Ji, F. Chen, L. Cao, and Y. Gao. Cross-modality microblog sentiment prediction via bi-layer multimodal hypergraph learning. IEEE Transactions on Multimedia, 2018, 21(4):1062-1075.
15. KI. Kim, F. Steinke, M. Hein . Semi-supervised regression using hessian energy with an application to semi-supervised dimensionality reduction. In Advances in Neural Information Processing Systems, 2009, pp. 979-987.
16. Y. Huang, Q. Liu, F. Lv, Y. Gong, and D. Metaxas. Unsupervised image categorization by hypergraph partition. IEEE Transactions on Pattern Analysis Machine Intelligence, 2011, 33(6):1266-1273.
17. X. Min, Y. Chen, S. Ge. Nonnegative matrix factorization with Hessian regularizer. Pattern Analysis and Applications, 2018,21(2):501-513.
18. S. Nene S, S. Nayar , H. Murase. Columbia object image library, Technical Report CUCS-005-96, 1996
19. J. Hull.A database for handwritten text recognition research. IEEE Transactions on Pattern Analysis and Machine Intelligence,1994, 16(5):550-554.
20. Y. Sun, S. Ding, Z. Zhang. et al. Hypergraph based semi-supervised support vector machine for binary and multi-category classifications. national Journal of Machine Learning and Cybernetics, 2022, 13:1369-1386
21. K. Kim, F. Steinke, M. Hein . Semi-supervised regression using hessian energy with an application to semi-supervised dimensionality reduction. In Advances in Neural Information Processing Systems, 2009, pp. 979-987.
22. A. Nene, S. K. Nayar and H. Murase. Columbia Object Image Library (COIL-100),Technical Report CUCS-006-96, February 1996.