Photo Semantic Understanding and Retargeting by a Noise-Robust Regularized Topic Model

Guifeng Wang, Luming Zhang, Yongbin Li, and Yichuan Sheng

***Abstract*—Retargeting aims at displaying a photo with an arbitrary aspect ratio, wherein the visually/semantically prominent objects are appropriately preserved and visual distortions can be well alleviated. Conventional retargeting models are built upon the visual perception of photos from a family of prespecified communities (*e.g.*, "portrait"), wherein the underlying community-specific features are not learned explicitly. Thus, they cannot appropriately retarget aerial photos, which contains a rich variety of objects with different scales. In this article, a novel aerial photo retargeting framework is designed by encoding the deep features from automatically detected Google Maps (https://www.google.com/maps) communities into a regularized probabilistic model. Specifically, we first propose an enhanced matrix factorization (MF) algorithm to calculate communities based on million-scale Google Maps pictures, for each of which deep feature is learned simultaneously. The enhanced MF incorporates label denoising, between-communities correlation, and deep feature encoding collaboratively. Subsequently, a probabilistic model called latent topic model (LTM) is designed that quantifies the spatial layouts of multiple Google Maps communities in the underlying hidden space. To alleviate the overfitting from Google Maps communities with imbalanced numbers of aerial photos, a regularizer is added into the LTM. Finally, by leveraging the regularized LTM, we shrink the test photo horizontally/vertically to maximize the posterior probability of the retargted photo. Comprehensive subjective evaluations and visualizations have demonstrated the advantages of our method. Besides, our calculate Google Maps communities are competitively consistent with the ground truth, according to the quantitative comparisons on the 2 M Google Maps photos.**

***Index Terms*—Aerial photo, deep feature, matrix factorization, probabilistic model, retargeting.**

I. Introduction

With the widespread availability of displayers in the past decade, retargeting has becoming a useful technique that displays an aerial photo on screens with different aspect ratios. For example, to design the wallpaper for a cell phone, we can adapt a high-resolution aerial photo cropped from Google Maps to a low-resolution cell phone screen. It is generally acknowledged that nonuniform scaling is suboptimal when the targeted aerial photo's aspect ratio is apparently different from the original one. Meanwhile, cropping performs unsatisfactorily if the visually/semantically salient regions are located dispersely. Aiming at a cross-resolution displaying technique, content-aware photo retargeting was proposed, focusing on optimally preserving visually/semantically important regions while shrinking the unimportant ones to reasonable scale. We have observed that, the existing content-aware retargeting [32], [34], [36], [37] algorithms are still frustrated to handle aerial photos due to the following challenges.

1) Conventional retargeting models are typically trained by utilizing well-composed aerial photos from a range of communities, wherein the community-specific deep features are not explicitly encoded. As exemplified in Fig. 1, for community "city," the salient objects are scattered around the aerial photo. Thus, the retargeting process is achieved similarly to uniform scaling. Comparatively, for community "airport," there are typically one or multiple aircrafts centered in the aerial photo. Therefore, the optimal retargeting should well preserve the center objects while maximally squeezing the rest background areas.
2) There are million-scale online aerial photo hosted by photo sharing websites, *e.g.*, Google Maps and Ope-

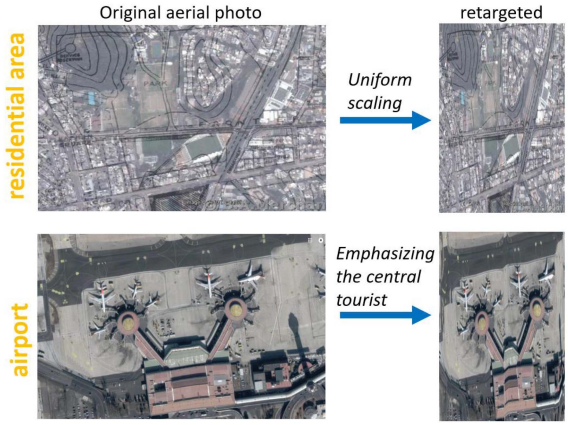

Fig. 1. Aerial photos from different communities are with different spatial layouts and thus should be shrunk differently.

Manuscript received 23 May 2022; revised 30 July 2022 and 8 September 2022; accepted 1 October 2022. Date of publication 22 February 2023; date of current version 11 April 2023. This work was supported in part by the Science and Technology Program of Jinhua, China (2020-1-004a), and in part by the Basic Public Welfare Research Project of Zhejiang Province (LGG19E050010). *(Corresponding authors: Guifeng Wang; Luming Zhang.)*

The authors are with the Key Laboratory of Crop Harvesting Equipment Technology of Zhejiang Province, Jinhua Polytechnic, Jinhua 321007, China (e-mail: guifengwang@zju.edu.cn; zglumg@zju.edu.cn; yongbingli221@sina.com; yichuansheng121@sina.com).

Digital Object Identifier 10.1109/JSTARS.2023.3247745

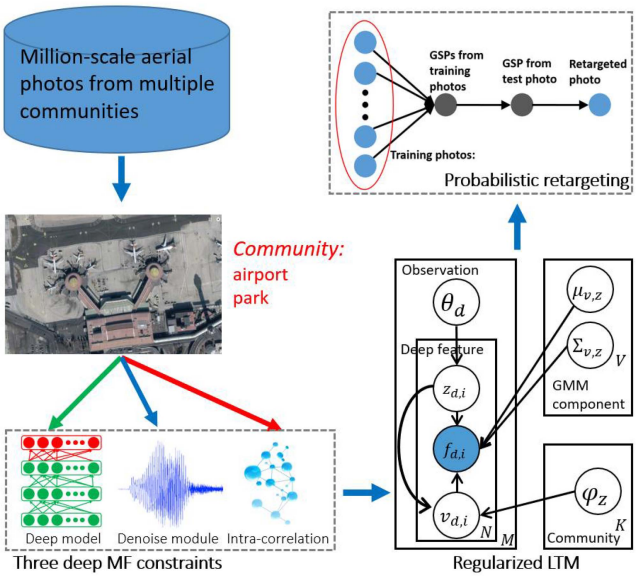

Fig. 2. Pipeline of our proposed community-aware photo retargeting.

nAerialMap.[1] It is feasible to employ aerial photos from them to learn a community-aware retargeting model. Actually, however, the Google Maps communities are manually built and maintained, which might be noisy. Ideally, we want a data mining system that can automatically detect communities. But designing such a system is difficult. Potential challenges include how to intelligently avoid the noisy community labels and how to exploit the intracorrelation between communities.

3) Each aerial photo website like Google Maps contains a rich variety of communities. Theoretically, learning a retargeting model that successfully represents the visual perceptual elements to multiple communities is a difficult task. Also, for Google Maps communities such "park" and "residential area" are somewhat relevant since they contains lots of "houses" photos. Contrastively, communities like "airport" and "intersection" are nearly irrelevant. This examples shows the importance of calculating the underlying hidden topics from multiple Google Maps communities. Moreover, some aerial photo communities typically contains too few aerial photos, which will lead to overfitting when training models training.

To overcome the aforementioned problems, a community-guided aerial photo retargeting is designed by encoding deep features from intelligently detected communities using a regularized latent topic model (LTM). The proposed LTM can be robustly learned in the presence of noisy image-level labels in the aerial photo set. Besides, the regularized term can make our LTM model has a high generalization ability toward categories with very few aerial photos. An overview of our pipeline is displayed in Fig. 2. Given a rich set of aerial photos, each represented by one or multiple communities, we propose an enhanced MF algorithm to derive the community label of each aerial photo. The MF seamlessly integrates three modules: community label denoising; intracorrelation between communities; and deep semantic encoding. Accordingly, an iterative algorithm is utilized to solve the MF problem. Afterward, to learn the latent topics from communities discovered by our MF and to handel overfitting, a regularized probabilistic topic model is formulated to quantify the styles from multiple Google Maps communities as a feature in the underlying hidden space. Finally, according to the learned feature, the aerial photo retrageting process is conducted based on a probabilistic model, where the test aerial photo is shrunk either horizontally or vertically to maximize the posterior probability. Extensive user studies on our compiled aerial photo set have demonstrated the competitiveness of our approach. Besides, quantitative comparisons have shown that the Google Maps communities discovery algorithm remarkably outperforms a series of counterparts.

Actually, these hidden communities are not visible compared to the massive-scale downloadable aerial photos. They are abstract concepts calculated using some data mining technique, by mimicking human visual perception and cognition. For humans, when they observe a huge number of aerial photos, they will perceptually categorize these aerial photos into multiple abstract concepts, such as "metropolises" and "industrial park." For our method, we proposed a novel regularized topic model to automatically discover the abstract concepts from the massive-scale aerial photos. The benefits are twofold:

1) there is no need to predefine the abstract concepts, which is highly challenging task based on the domain experiences;
2) we can tune the number of abstract concepts and check whether our wanted abstract concepts are discovered.

This can be used to facilitate different applications.

The main contributions of our work are given as follows. First, a noise-tolerant MF is proposed that simultaneously combine community label denoising, intracorrelation between communities, and deep semantic encoding. Second, a novel probabilistic topic model is designed that effectively calculates the features of aerial photos from different Google Maps communities, wherein the overftting problem can be optimally addressed. Third, an in-depth experimental evaluation on 2 M aerial photos cropped from Google Maps is conducted to evaluate our method comprehensively.

The rest of this article is organized as follows. Section II reviews the related work in the past decade. Section III elaborates the three important modules in our retargeting framework: a noise-refined deep MF algorithm for Google Maps communities discovery; a regularized LTM for modeling the distribution of aerial photos from multiple communities; and a probabilistic model that maximizes the posterior probability of the retargeted aerial photo. Experimental validations in Section VI verified the advantage of our proposed framework. Finally, Section V concludes this article.

II. Related Work

Our retargeting model is motivated by two hot directions in image processing and data engineering: content-aware photo retargeting and community learning.

[1] openaerialmap.org

A. Content-Aware Photo Retargeting

There are tens of retargeting algorithm in image processing domain, where a few representative ones are reviewed as follows. Avidan et al. [3] formulated image retargeting as dynamic programming-based seam detection, based on which a gradient energy is utilized to indicate the pixels' significance. Bubinstein et al. [34] proposed an energy optimization objective to enhance Avidan et al.'s work. As an upgraded version of seam carving, Pritch et al. [32] discretely abandoned the duplicated visual patterns from each segmented regions. Wolf et al. [39] proposed to optimally combine less important pixels to achieve invisible visual distortion, which is propagated along the shrinking direction. In [36], Sun et al. proposed a retargeting algorithm that effectively produces thumbnails from input images. In [13], Guo et al. designed a robust photo retargeting algorithm by leveraging saliency-based mesh parametrization. Lin et al. [23] proposed a patch-based retargeting by preserving the shapes with both visually attractive regions and geometric structures. By constructing a latent space combining a set of operators [35], Rubinstein et al. proposed a retargeting scheme by optimizing the operation path in the latent space. Wang et al. [38] proposed a video retargeting method by conducting cropping and wrapping iteratively. The cropping removes the temporally repeated contents while the warping leveraging homogeneous regions to reduce deformations and maintain motional feature. In [30], Panozzo et al. retargeted photos in the space constructed by various deformation operations. Recently, Castillo et al. [7] analyzed the influences of human gaze behavior on retargeting, based on the comprehensive experimental evaluations on the RetargetMe [33]. Noticeably, the aforementioned retargeting models can only handle pictures captured by consumer cameras. They may not successfully retarget aerial photos that are captured by high precision optical sensors installed on satellites.

B. Learning Communities Technique

The objective of learning communities is to discover the inherent sophisticated structures from massive-scale networks. Theoretically, one community can be deemed as a cluster of densely distributed vertices (each representing an aerial photo from Google Maps in our context), which are loosely connected to the other clusters. The communities discovery problem has been studied deeply and extensively in machine learning. Subtopics include mining multiple overlapping communities proposed by Gregory [12] and Zhang et al. [44], community mining from bipartite graphs [31], and jointly encoding side information and network structure for communities discovery [41] work. Lancichinetti et al. [21] theoretically and empirically compared two communities discovery algorithms by employing a rich set of baseline graphs. In [22], the authors pointed out that deriving the underlying communities structure from massive-scale networks is extremely challenging, since many networks with complicated structures are optimized locally. Yoshida [42] proposed to mine communities based on Internet-scale social networks by taking advantage of the complicated graph geometry. In [10], the authors proposed to categorize Boolean vectorial feature into multiple communities, *i.e.*, the vertices' features are labeled by multiple communities. By representing vertex memberships from overlapping communities, Yang and Leskovec [40] formulated an objective function to search overlapping communities from multiple networks. We notice that the aforementioned techniques cannot well handle the possibly contaminated community labels in the model training stage. Even worse, the inherent characteristics among multiple communities is not explored.

III. Our Method

A. MF Under a Deep Framework

Practically, each aerial photo is associated with one or multiple communities, as exemplified in Fig. 1. It is significant to exploit the community information during aerial photo retargeting. Herein, for label matrix containing the community labels $\mathbf{C} \in \mathbb{R}^{C\times N}$, where N counts the aerial photos and C counts the unique labels, our proposed MF framework seeks to characterize it using the product of two factor matrices $\mathbf{P} \in \mathbb{R}^{H\times C}$ and $\mathbf{Q} \in \mathbb{R}^{H\times N}$, *i.e.*,

$$\min_{\mathbf{P},\mathbf{Q}} \frac{1}{2}||\mathbf{C}-\mathbf{P}^T\mathbf{Q}||_F^2 + \frac{\tau_1}{2}||\mathbf{P}||_F^2 + \frac{\tau_2}{2}||\mathbf{Q}||_F^2 \tag{1}$$

where the overfitting problem can be well handled by the pairwise regularizers; τ_1 and τ_2 are pairwise positive parameters between zero and one; factor matrices $\mathbf{P}$ and $\mathbf{Q}$, respectively, denote the basis matrix and hidden community matrix calculated by N aerial photos.

Noticeably, the traditional MF simply discover the clues from communities. This cannot optimally avoid the negative effects from the noisy community labels. At the same time, the complicated relationships between Google Maps communities fail to be captured. Furthermore, semantically modeling the hidden structure of an unknown aerial photo is also difficult. To solve these problems, an enhanced matrix factorization is developed.

1) Denoising: The labels of aerial photos are annotated by humans typically, which is usually noisy and even incomplete. In our work, a novel label denoising component $\Phi(\mathbf{C},\mathbf{Y})$ is proposed. The label noise refinement module is directly incorporated into the previous community matrix. This is conducted by calculating the relevant community labels that transfer information between multiple aerial photos and their labels. In machine learning domain, it is apparently that the l_1 norm can produce a high tolerance to label noises [18]. In this way, a novel l_1 norm is applied here

$$\Phi(\mathbf{C},\mathbf{Y}) = \alpha||\mathbf{C}-\mathbf{Y}||_1 \tag{2}$$

Herein, $\mathbf{Y} \in \mathbb{R}^{C\times N}$ denotes the matrix capturing the annotated labels obtained from a rich set of aerial photos.

2) Correlation Encoding: For our proposed MF that can robustly handle label noises, it is necessary to optimally encode the inherent relationships between a set of Google Maps communities. This is because the underlying hidden distribution of Google Maps communities is an informative feature for exploiting these communities. Herein, we characterize the underlying correlations by leveraging the operation of inner product. Formally speaking, such inner product operation can

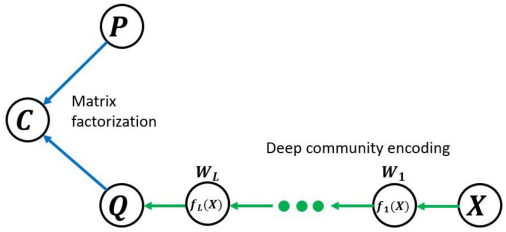

Fig. 3. Illustration of our designed deep community encoding model.

be represented as

$$\Gamma = \frac{\beta}{2}||\mathbf{S} - \mathbf{Q}^T\mathbf{Q}||_F^2 \tag{3}$$

where $\mathbf{S} \in \mathbb{R}^{C\times C}$ denotes a matrix that is symmetric. It can encode the differences between C Google Maps communities. In detail, the ijth each element can be computed via

$$\mathbf{S}_{ij} = \exp\left(-\frac{d_{JS}(\theta_i||\theta_j)}{2\tau^2}\right) \tag{4}$$

where θ_i represents the feature distribution of deep convolutional neural network (CNNs) [20] calculated from the entire aerial photos coming from the each community. Simultaneously, $d_{JS}(\cdot,\cdot)$ denotes the JS divergence [11].

3) Deep Community Encoding: In order to embed an unseen aerial photo, $\mathbf{W} \in \mathbf{R}^{H\times R}$ is deployed to convert aerial photos' deep features into the H-dimensional hidden community features capturing the styles of different aerial photos, *i.e.*, $\mathbf{Q} = \mathbf{WX}$. Herein, matrix $\mathbf{X} \in \mathbb{R}^{R\times N}$ represents the R-dimensional deep features extracted by leveraging the N aerial photos.

Noticeably, simply mapping the extracted deep features into the underlying hidden community space might be imperfect owing to the insurmountable semantic gap. Thanks to the excellent performance of hierarchical features [14], [20], an hierarchial learning framework is designed to calculate the underlying hidden community representation (as elaborated in Fig. 3). Such component can be optimally integrated into the aforementioned MF architecture for Google Maps communities mining. Moreover, a deep CNN is designed that includes convolutions as well as pooling functions that end-to-end calculate deep features from aerial photo regions. For the CNN having L layers, each layer $f_l(x_i)$'s output can be obtained in the following way:

$$f_l(x_i) = \phi(\mathbf{W}_l f_{l-1}(x_i) + \epsilon_l), l = 1, \ldots, L \tag{5}$$

where $\phi(\cdot)$ describes the function that activates; and $\mathbf{W}_l$ capture the weight matrix from the ith layer, which maps deep feature x_i to the corresponding deep representation y_i. Notably, we assume a linear projection between the input and output in each layer of our deep model. $f_l(\cdot)$ and ϵ_l, respectively, means the deep feature from the lth layer and the bias. Herein, our designed deep model is pretrained by leveraging the well-known ImageNet [20]. Specifically, the output from the first layer $\mathbf{W}f_L(\mathbf{X})$ represents the hidden community features extracted from N aerial photos. In our implementation, following [20], we set $L = 7$.

We integrate these noise reduction, correlation, and deep community features into the traditional MF, based on which the following objective function can be obtained:

$$\min_{\mathbf{C},\mathbf{P},\mathbf{W},\Theta} \frac{1}{2}||\mathbf{C} - \mathbf{P}^T\mathbf{W}f_L(\mathbf{X})||_F^2 + \alpha||\mathbf{C} - \mathbf{Y}||_1 + \frac{\beta}{2}||\mathbf{S} - f_L^T(\mathbf{X})\mathbf{W}^T\mathbf{W}f_L(\mathbf{X})||_F^2 + \mathcal{R}(\Theta, \mathbf{W}, \mathbf{P}), \text{ s.t., } \mathbf{C} \in \{0,1\}^{C\times N} \tag{6}$$

where Θ contains the parameters of the deep model; and $\mathcal{R}(\cdot)$ denotes the regularization term toward all the inherent parameters.

The aforementioned objective function is nonconvex toward all the parameters. Thus, we propose to solve it by an iterative algorithm as detailed in drive.[2]

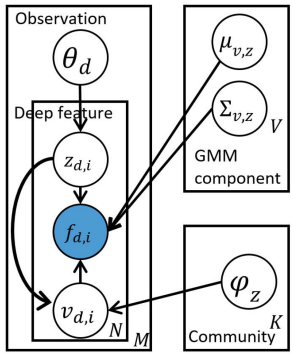

Fig. 4. Illustration of our proposed GMM-LTM.

B. Regularized LTM

After labeling the communities to the million-scale aerial photos, we then characterize the distribution by discovering the hidden topics by exploiting a rich set of communities. The latent topics are informative clues for retargeting since some Google Maps communities are closely correlated in semantics (*e.g.*, "rooftop," and "house"). Therefore, it is significant to combine these highly correlated communities.

In our implementation, a Gaussian latent topic model is designed and can be mathematically described in Fig. 4. For a deep feature calculated using aerial photo d associated with Google Maps community tag z as well as the corresponding Gaussian component v, the overall distribution is given as

$$p(\mathbf{F}|\Upsilon) = \prod_{d=1}^{M}\prod_{i=1}^{N}\sum_{z=1}^{K}\sum_{v=1}^{V} p(f_{d,i}|\mu_v, \Sigma_v)p(z_{d,i}|\theta_d)p(v_{d,i}|\varphi_z) \tag{7}$$

where $p(f_{d,i}\mu_{d,z}, \Sigma_{d,z})$ denotes the multivariate distribution wherein μ and Σ, respectively, denote the mean and variance of the Gaussian components; $p(z_{d,i}|\theta_d)$ and $p(v_{d,i}|\varphi_z)$ denote pairwise multinomial distributions; and $\Upsilon = \{\mu, \Sigma, \theta, \varphi\}$.

Practically, the numbers of aerial photos within different communities are also different. A few Google Maps communities contain very few aerial photos. Thus, the overfitting will be generated when training the model. Therefore, a regularizer

[2]https://drive.google.com/file/d/1PMJWmsWDRPvlM6WqmsYXMqA1j0NQ6YFn/view?usp=sharing

is designed to upgrade our designed Gaussian LTM. Herein, the inherent parameters are optimized by maximizing the below objective function, that is,

$$q(\Upsilon|\Upsilon^t,\Upsilon_g^t) \triangleq p(\mathbf{F}|\Upsilon)+p(\mathbf{F}|\Upsilon^t)+\rho * p(\mathbf{F}|\Upsilon^t,\Upsilon_g^t) \quad (8)$$

where ρ is a positive weight reflecting the regularizer's significance. $p(\mathbf{F}|\Upsilon^t)$ and $p(\mathbf{F}|\Upsilon^t,\Upsilon_g^t)$ are calculated as follows:

$$p(\mathbf{F}|\Upsilon^t)=\prod_{d=1}^{M}\prod_{i=1}^{N}\sum_{z=1}^{K}\sum_{v=1}^{V}p(z,v|\Phi^t)*\log\frac{p(f_{d,i},z,v|\Upsilon)}{p(f_{d,i},z,v|\Upsilon^t)} \quad (9)$$

$$p(\mathbf{F}|\Upsilon^t,\Upsilon_g^t)=\prod_{d=1}^{M}\prod_{i=1}^{N}\sum_{z=1}^{K}\sum_{v=1}^{V}p(z,v|\Phi_g^t)*\log\frac{p(f_{d,i},z,v|\Upsilon)}{p(f_{d,i},z,v|\Upsilon_g^t)}. \quad (10)$$

Thereafter, our proposed regularized LTM is written as

$$\hat{p}(\mathbf{F}|\Upsilon)=p(\mathbf{F}|\Upsilon)+\rho * p(\mathbf{F}|\Upsilon^t,\Upsilon_g^t). \quad (11)$$

Optimization: Formally, (8) is represented as

$$q(\Upsilon|\Upsilon^t,\Upsilon_g^t)\propto E_{p_r(z,v|\mathbf{F},\Upsilon^t,\Upsilon_g^t)} * \log p(\mathbf{F},z,v|\Upsilon) \quad (12)$$

where $p_r(z,v|\mathbf{F},\Upsilon^t,\Upsilon_g^t)=\frac{p(z,v|\mathbf{F},\Upsilon^t)+\rho * p(z,v|\mathbf{F},\Upsilon_g^t)}{1+\rho}$ represents the inherent structure with respect to the parameters.

Thereafter, we propose an updated EM to iteratively derive the weights in (12). During the E-step, according to the existing parameters, the posterior provability can be computed as follows:

$$l_{z,v}=p_r(z,v|\mathbf{F},\Upsilon^t,\Upsilon_g^t). \quad (13)$$

During M-step, we integrate the Lagrange multipliers. Accordingly, we calculate the parameters by optimizing the following objective function:

$$\Upsilon^{t+1}=\arg\max_{\Upsilon} q(\Upsilon|\Upsilon^t,\Upsilon_g^t)+\gamma_1\sum_{d=1}^{M}\left(1-\sum_{z=1}^{K}\theta_{d,z}\right)+\gamma_2\sum_{z=1}^{K}\left(1-\sum_{v=1}^{V}\phi_{z,v}\right). \quad (14)$$

By solving the objective function (14), we can obtain

$$\theta_{z,v}^{t+1}=\frac{\sum_{v=1}^{V}l_{z,v}}{\sum_{z=1}^{K}\sum_{v=1}^{V}l_{z,v}} \quad (15)$$

$$\varphi_{z,v}^{t+1}=\frac{\sum_{d=1}^{M}l_{z,v}^{d}}{\sum_{z=1}^{K}\sum_{v=1}^{V}l_{z,v}^{d}} \quad (16)$$

$$\theta_{g}^{t+1}=\frac{\exp\left(\sum_{d=1}^{M}\sum_{z=1}^{K}\log\theta_{d,z}^{t+1}/MK\right)}{\exp\left(\sum_{d=1}^{M}\sum_{z^*=1}^{K}\log\theta_{d,z^*}^{t+1}/MK\right)}. \quad (17)$$

According to (13), (15), (16), and (17), our designed EM is carried out iteratively.

C. Probabilistic Model for Retargeting

In computer vision community, aerial photo visual perception is subjective, different photographers might have different opinions toward the same aerial photo. To handle this problem during the retargeting process, we attempt the encode the visual perception experiences of Google Maps users from the test aerial photo's community. Specifically, we model the styles of aerial photos in the various Google Maps communities mined by us. In our implementation, we adopt the aforementioned regularized LTM to characterize the distribution of deep features [calculated by (5)] from aerial photos inside multiple Google Maps communities.

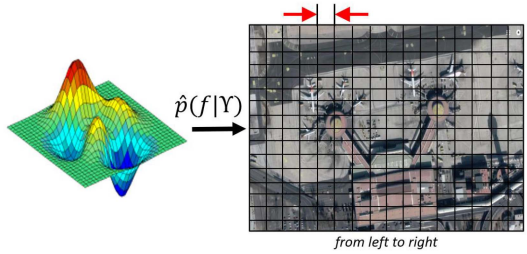

Fig. 5. Illustration of our aerial photo retargeting by grid shrinking.

Apparently, the retargeted aerial photo should be similarly perceived to the training aerial photos from multiple communities. For a new aerial photo, we obtain its deep feature, based on which the probability of each grid is calculated. During shrinking, to alleviate using triangle mesh, which inevitably produces distortions in triangle orientations, grid-guided shrinking scheme is utilized. More specifically, the test aerial photo is evenly divided into equal-sized grids. Based on this, we calculate the horizontal weight of grid ϕ as

$$w_h(\phi)=\max_{f}\hat{p}(f|\Upsilon). \quad (18)$$

It is noticeable that, the shrinking process is conducted one-by-one (from left to right as displayed in Fig. 5). For each shrinking step, a temporary retargeted aerial photo is produced. In (18), f is the deep feature corresponding to the current test aerial photo during shrinking, and probability $\hat{p}(f|\Upsilon)$ follows the proposed regularized LTM as detailed in (11).

After calculating each grid's horizontal weight, a normalization is conducted as follows:

$$\bar{w}_h(\phi_i)=\frac{w_h(\phi_i)}{\sum_i w_h(\phi_i)}. \quad (19)$$

Assuming that the retargeted aerial photo has a size of $W\times H$, the horizontal and vertical dimensions of the ith grid is squeezed to $[W\cdot\bar{w}_h(\phi_i)]$ and $[H\cdot\bar{w}_v(\phi_i)]$, respectively. Herein, $[\cdot]$ rounds a real number. As exemplified in Fig. 5, the foreground aircrafts are visually/semantically salient. And thus they are kept in the retargeted aerial photo without shrinkage. Comparatively, the backgrounds are less semantically important. In this way, they will be compressed both horizontally and vertically.

Based on the descriptions in this section, an overview of our aerial photo retargeting is presented in Algorithm 1.

Algorithm 1: Community-Aware Aerial Photo Retargeting using Deep Noise-refined MF.

input: Million-scale aerial photos with community labels, parameters α, β, τ_1, τ_2 ρ, C and a test aerial photo;
output: Matrices $\mathbf{W}$, $\mathbf{C}$, and the retargeted aerial photo;

1) Use our deep noise-robust MF to discover a series of Google Maps communities, and calculate the deep feature for each aeiral photo;
2) Use the regularized probabilistic model to calculate the distribution of aerial photos in each of the C communities;
3) Grid-based aerial photo shrinking based on the posterior probability calculated from the probabilistic model, and output the retargeted aerial photo.

IV. Experimental Evaluation

Herein, we validate the performance of our method by leveraging three experiments. The first experiment compares our method with a series of popular photo retargeting algorithms. Subsequently, we carefully test each module in our proposed aerial photo retargeting pipeline: deep noise-refined MF; the regularized LTM; and the probabilistic model for retargeting. Third, we show the influence of important parameters on retargeting.

The entire aerial photos for experimentation are collected from Google Maps. The entire aerial photo set involves more than 2 M samples cropped from multiple well-known Google Maps communities. There are approximately $90\,000 \sim 120\,000$ photos crawled from different continents throughout the world in each community. Herein, we randomly select 50% aerial photos from each community, in order to learn our enhanced MF framework. Based on this, we employ 80 aerial photos as well as the standard RetargetMe [33] to evaluate the retargeting performance.

A. Comparative Study and Analysis

1) Retargeting Performance: We evaluate our designed retargeting model by comparing with many baseline methods, that is, seam carving (SC) and its enhanced version (ISC) [3], optimized scale-and-sketch (OSS) [37], and saliency-based mesh parametrization (SMP) [13]. We first report the aerial photo retargeting performance in Fig. 6. As can be seen, the following observations are obtained. First, our pro- posed retargeting model nicely encodes those semantically significant targets inside different aerial photos, like the faces and the architecture. This observation reflects that our probabilistic retageting model can obtain the spatial layouts of well-composed images. As shown, the foreground prominent objects are salient in each image. In contrast, the baseline models sometime strongly shrink the key objects and produce observable visual distortion. Next, the key objects as well as the surroundings are nicely arranged in the generated photos. As an example, the intersection, the gymnasium, and the surroundings are harmonically distributed. The pairwise barrels and palette, and their complicated spatial configurations, can be optimally kept. Last but not least, the boat and the architectures are the key to the retargeted picture. These objects are perfectly preserved.

Fig. 6. Aerial photos retargeted by different algorithms.

Additionally, we compare the retargeting performance on generic photos from RetargetMe [33]. As can be seen from Fig. 7, our method retargets aerial photos more aesthetically pleasing. The central visually/semantically salient objects are well preserved with less shrinkage. Moreover, our method produces retargeted photos with least perceptual visual distortions.

Afterward, a comprehensive user study is conducted to make a comparison toward a series of retargeting algorithms. Among these, 40 volunteers recruited from our Computer Sciences Department are participated. For each volunteer, the retargeted as well as the reference aerial photos, and solely the retargeted aerial photo not including the reference one. We follow the setup in [33], wherein the agreement coefficient [19] is calculated in order to quantify volunteers' opinions on aerial photos retargeted by various techniques. Herein, a relatively low score quantifies the difficulty in making a decision. Meanwhile, the calculated agreement coefficients over all the experimental aerial photos are reported in Fig. 8. It is noticeable that, in Fig. 8(a), the agreement parameter decreases sharply since there is no reference aerial photo displayed. As shown, volunteers made strong agreements on attributes like "face/people" and "symmetry." This is because that human face is visually important in visual perception and symmetry is an implicitly perceptual attribute. As can be seen from Fig. 8(b), users hold the opinion that our approach significantly outperforms its counterparts on attributes "face/people" and "texture." Comparatively, for the rest attributes, our method slightly surpasses its counterparts.

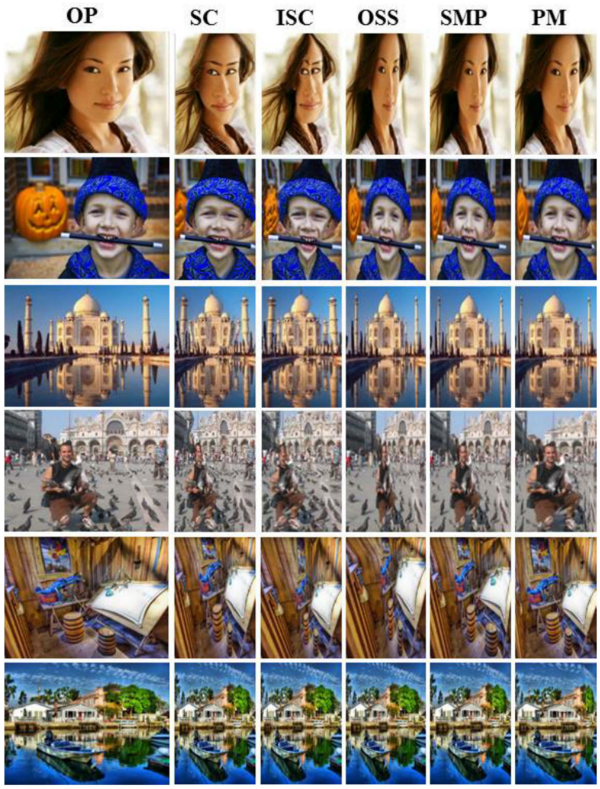

Fig. 7. RetargetMet [33] photos retargeted by different algorithms.

	Lines/ edges	Faces/ people	texture	Fore. objects	Geometric structure	Symmetry	Aggregate
μ (with ref.)	0.112	0.194	0.196	0.157	0.126	0.214	0.104
μ (without ref.)	0.934	0.187	0.196	0.145	0.116	0.212	0.991

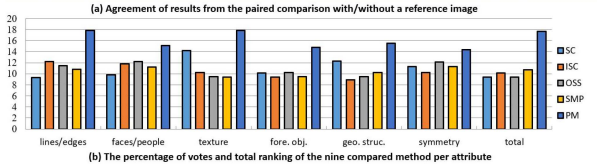

Fig. 8. Quantitative analysis of the five baseline retargeting algorithms.

2) Visual Perception Encoding: To further validate the competitiveness of our retargeting model, we testify the visual perceptual descriptiveness of our approach. We generate nearly 1000 candidate subregions by tuning the central translation [43] with a 10-pixel interval. Herein, the aspect ratio is fixed to the same as that of the previous aerial photo. Thereafter, $\hat{p}(\mathbf{F}|\Upsilon)$ is utilized to quantify the perception of each sub region in an aerial photo. Afterward, the highest scoring one is calculated. Accordingly, ten testing aerial photos combined with their highest/lowest quality subregions in Fig. 9. We also create a perceptual map by calculating the perceptual quality toward all the patches corresponding to the largest score of the subregions inside it. As can be seen from Fig. 9, the subregions with different quality attributes are apparently different, based on the calculated saliency map. Overall, the following conclusions can be obtained.

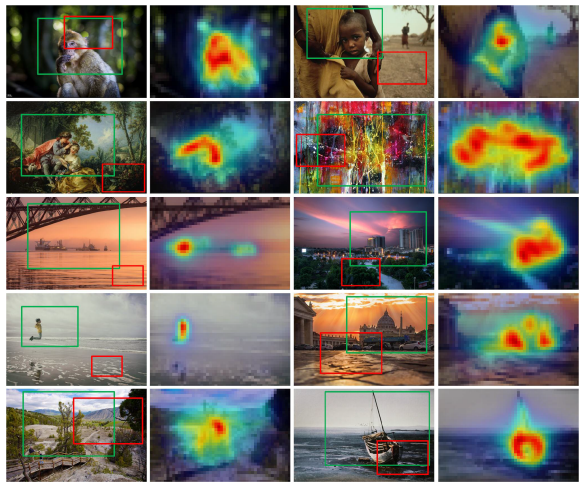

Fig. 9. Perceptual quality evaluation (the differently colored windows respectively denote subregions with the different quality attributes.).

1) The top salient regions predicted by our model usually describe those foreground semantic objects, *e.g.*, the humans and mosque. It shows that maintaining the the central semantically important objects is the key for visual composition. When observing the highest/lowest perceptual quality subregion in Fig. 9, it is noticeable that the highest ranking subregions cover those foreground objects. In contrast, the lowest scoring subregions typically contain foreground objects incompletely, that is, the face and yacht.
2) The designed method not only well reflects the semantically important targets but also can effectively suggest optimally composed subregion. For the highest scoring subregions Fig. 9, the central key objects are typically located near the diagonal line or surrounded by a set of objects (like the mosque and trees).
3) Our designed method can nicely capture aerial photos with a rich set of styles, such as portraits, landscape, architecture, and even abstract painting. This is because the proposed deep noise-fined MF learns a descriptive compositional descriptor, and the regularized LTM can optimally encode various image styles from multiple communities.

B. Component-Wise Evaluation

Generally, there are two main parts in our proposed aerial photo retargeting pipeline: the deep noise-refined MF and regualized LTM. Herein, we carefully evaluate the usefulness of these two modules.

For the first component, we compare it with multiple famous communities discovery techniques. In particular, the multiple benchmark techniques are as follows:

1) mixed membership stochastic block (MMSB) [2];
2) block-LDA [4];
3) K-means clustering (KC) [27];
4) hierarchical clustering (HC) [17];
5) link clustering (LC) [1];

TABLE I
BER SCORES OF DIFFERENT GOOGLE MAPS COMMUNITIES DISCOVERY ALGORITHMS

GM group	MMSB	BLDA	KC	HC	LC	CP	LRE	MAC	Gregory	Zhang	Yang	Ours(ICL)	Ours
Forest	0.4552	0.4176	0.4885	0.3009	0.4442	0.4135	0.3251	0.4024	0.3645	0.3782	0.4126	**0.4789**	0.4715
Aircraft	0.5574	0.5264	0.5437	0.4052	0.5223	0.4655	0.4237	0.4976	0.4668	0.4779	0.5141	0.5583	**0.5889**
Road	0.4084	0.3768	0.3549	0.3458	0.3656	0.3927	0.3565	0.3649	0.3587	0.3868	0.4035	**0.4315**	0.4137
Palace	0.6416	0.6157	0.5524	0.5956	0.5585	0.5493	0.6264	0.5762	0.5279	0.5359	0.5762	0.6537	**0.6628**
Sea	0.3259	0.3287	0.3074	0.3486	0.3169	0.3365	0.3779	0.3961	0.3689	0.3781	0.3978	**0.4113**	0.4094
Railway	0.7187	0.6886	0.6478	0.6584	0.6768	0.6426	0.6229	0.6437	0.6484	0.6779	0.7081	0.7205	**0.7258**
River	0.5994	0.6095	0.5474	0.5693	0.5471	0.5789	0.5674	0.5876	0.5484	0.5849	0.6053	**0.6194**	0.6115
Factory	0.5268	0.5184	0.4972	0.5153	0.4771	0.4986	0.4632	0.4966	0.5074	0.4594	0.5157	**0.5395**	0.5351
Tall building	0.6272	0.6132	0.6041	0.6052	0.5534	0.5743	0.5865	0.6051	0.6063	0.6048	0.6274	0.6243	**0.6315**
Residential	0.5964	0.6179	0.5533	0.5854	0.5742	0.5551	0.5935	0.5444	0.5758	0.5679	0.6095	**0.6246**	0.6183
Intersection	0.2394	0.2472	0.1961	**0.3267**	0.2366	0.2049	0.1786	0.2488	0.2074	0.1961	0.2043	0.2217	0.2243
Soccer field	0.6154	0.5961	0.5587	0.5556	0.5792	0.5643	0.5584	0.5742	0.5688	0.5761	0.5945	0.6175	**0.6367**
Bridge	0.7274	0.7152	0.6474	0.6763	0.7178	0.6475	0.7064	0.6619	0.6789	0.6823	0.7165	**0.7394**	0.7343
Park	0.6486	0.6179	0.6068	0.6262	0.5784	0.5968	0.6072	0.6493	0.6416	0.6525	0.6419	0.6521	**0.6583**
Farmland	0.3265	0.2949	0.2675	0.2721	0.2746	0.3016	0.3145	0.3165	0.3158	0.3242	0.3387	**0.3451**	0.3445
Townlet	0.6442	0.6262	0.6056	0.6162	0.5945	0.5778	0.5785	0.5742	0.5974	0.5956	0.6085	**0.6457**	0.6262
Racetrack	0.7289	0.6945	0.6572	0.6845	0.6929	0.6845	0.6825	0.7063	0.7057	0.6936	0.7058	0.7314	**0.7342**
Playground	0.3974	0.3742	0.3858	0.3584	0.4151	0.4045	0.3367	0.3774	0.3848	0.3824	0.4076	**0.4229**	0.4197
Wharf	0.4342	0.4065	0.4247	0.4244	0.3963	0.4075	0.3745	0.3256	0.3634	0.3595	0.4041	0.4372	**0.4418**
Valley	0.5545	0.5142	0.5321	0.5126	0.4835	0.5015	0.4984	**0.5564**	0.4543	0.4794	0.4865	0.4512	0.4913
Average	0.5387	0.5200	0.5007	0.4991	0.5001	0.4949	0.4889	0.5053	0.4946	0.4997	0.5240	0.5463	**0.5490**

The bold values represent the best performer.

TABLE II
COMPARATIVE STANDARD ERROR OF BER SCORES OF DIFFERENT GOOGLE MAPS COMMUNITIES DISCOVERY ALGORITHMS

GM group	MMSB	BLDA	KC	HC	LC	CP	LRE	MAC	Gregory	Zhang	Yang	Ours(ICL)	Ours
Forest	0.0861	0.0645	0.0717	0.0633	0.0725	0.0621	0.0854	0.0675	0.0732	0.0671	0.0716	0.0637	0.0518
Aircraft	0.0636	0.0657	0.0716	0.0842	0.0749	0.0624	0.0749	0.0656	0.0592	0.0641	0.0548	0.0659	0.0427
Road	0.0651	0.0774	0.0549	0.0642	0.0353	0.0758	0.0663	0.0742	0.0473	0.0562	0.0675	0.0436	0.0622
Palace	0.0668	0.0632	0.0552	0.0636	0.0761	0.0632	0.0784	0.0661	0.0741	0.0668	0.0652	0.0735	0.3416
Sea	0.0635	0.0564	0.0613	0.0732	0.0651	0.0534	0.0664	0.0584	0.0635	0.0652	0.0589	0.0622	0.0418
Railway	0.0662	0.0649	0.0761	0.0623	0.0721	0.0741	0.0661	0.0632	0.0782	0.0652	0.0523	0.0651	0.0506
River	0.0762	0.0684	0.0846	0.0658	0.0736	0.0572	0.0668	0.0621	0.0582	0.0673	0.0569	0.0614	0.0468
Factory	0.0669	0.0726	0.0868	0.0753	0.0684	0.0756	0.0629	0.0751	0.0864	0.0618	0.0645	0.0834	0.0729
Tail building	0.0675	0.0761	0.0775	0.0642	0.0736	0.0556	0.0649	0.0558	0.0635	0.0656	0.0867	0.0615	0.0508
Residential	0.0652	0.0449	0.0572	0.0781	0.0662	0.0586	0.0656	0.0666	0.0632	0.0558	0.0669	0.0747	0.0423
Intersection	0.0674	0.0761	0.0674	0.0795	0.0643	0.0712	0.0767	0.0546	0.0659	0.0814	0.0732	0.0529	0.0344
Soccer field	0.0685	0.0584	0.0693	0.0786	0.0648	0.0742	0.0823	0.0724	0.0796	0.0691	0.0747	0.0512	0.0421
Bridge	0.0784	0.0684	0.0569	0.0764	0.0879	0.0758	0.0613	0.0767	0.0559	0.0774	0.0891	0.0522	0.0431
Park	0.0684	0.0764	0.0648	0.0557	0.0749	0.0875	0.0585	0.0683	0.0664	0.0551	0.0664	0.0559	0.0414
Farmland	0.0661	0.0774	0.0662	0.0442	0.0562	0.0659	0.0561	0.0674	0.0533	0.0632	0.0681	0.0675	0.0453
Townlet	0.0735	0.0634	0.0722	0.0861	0.0657	0.0686	0.0735	0.0712	0.0559	0.0689	0.0668	0.0732	0.0641
Racetrack	0.0854	0.0611	0.0763	0.0751	0.0616	0.0752	0.0558	0.0632	0.0416	0.0625	0.0732	0.0727	0.0511
Playground	0.0762	0.0753	0.0684	0.0724	0.0682	0.0586	0.0661	0.0535	0.0642	0.0712	0.0753	0.0629	0.0414
Wharf	0.0895	0.0945	0.0765	0.0664	0.0785	0.0694	0.0651	0.0721	0.0753	0.0659	0.0647	0.0612	0.0539
Valley	0.0656	0.0635	0.0759	0.0754	0.0623	0.0525	0.0636	0.0742	0.0751	0.0831	0.0664	0.0675	0.0515

6) clique percolation (CP) [29];
7) low-rank embedding (LRE) [42];
8) multiassignment clustering (MAC) [10].

Herein, three overlapping-guided communities discovery techniques are adopted for empirical study. For these employed techniques, we set the number of Google Maps communities by following the well-known BIC criterion. Furthermore, the integrate complete likelihood technique [5] is leveraged for evaluating the performance of our proposed technique. The entire data possibility is built upon the parameters learned from the expectation maximization (EM).

Following the comparative study [16], the BER [9] and F_1 scores [28] are calculated in our experiment. As shown in Tables I–IV, we can obtain the following conclusions.

1) For the entire 20 Google Maps clusters, in 18 Google Maps circles, the best performance is achieved by our method. This is because the internal BER and F_1 scores reach the top. The results show that the advantage of our method in handling noisy aerial photo labels, by employing the l_1 norm to simultaneously tackle contaminated aerial photo visual/semantic tags and the cluster tags. Nevertheless, the other cluster discovery techniques cannot well handle the noisy community tags effectively.
2) For Google Maps clusters reflecting detailed concepts, they are likely to be observed highly probably. Comparatively, for clusters capturing the abstract concepts, it is noticeable the all the communities discovery techniques are ineffective. This is because, aerial photos with abstract meanings usually are with a fixed appearance. In this way, the produced visual descriptors cannot well capture them.

To evaluate the performance of probabilistic model, two different setups are utilized. In the first place, our designed probabilistic model is replaced by the traditional Gaussian mixture model, the PGLSA [15], and the LDA [6], respectively. Also, we make a comparison with the LSTM by leveraging the categorization EM [8]. Afterward, the regularizer encoded in proposed LTM is abandoned by us. To validate the usefulness of these probabilistic topic models, the ground-truth distribution

TABLE III
COMPARATIVE F_1 SCORES OF DIFFERENT GOOGLE MAPS COMMUNITIES DISCOVERY ALGORITHMS

GM group	MMSB	BLDA	KC	HC	LC	CP	LRE	MAC	Gregory	Zhang	Yang	Ours(ICL)	Ours
Forest	0.2354	0.2463	0.3249	0.1942	0.1731	0.2742	0.1745	0.1942	0.1942	0.2138	0.2242	0.2825	**0.3283**
Aircraft	0.4134	0.3768	0.3446	0.2484	0.4026	0.2975	0.2785	0.2946	0.2968	0.2945	0.2951	**0.4167**	0.3075
Road	0.2375	0.2586	0.2145	0.1747	0.1946	0.2686	0.1875	0.2079	0.2158	0.2293	0.2369	**0.2864**	0.2747
Palace	0.5165	0.4794	0.4254	0.4292	0.3135	0.3937	0.4686	0.4135	0.4246	0.4384	0.4724	0.5071	**0.5243**
Sea	0.1942	0.2024	0.1841	0.2143	0.2092	0.2174	0.2342	0.2482	0.2135	0.2121	0.2384	**0.2587**	0.2559
Railway	0.6252	0.5667	0.6042	0.5876	0.6161	0.4589	0.4786	0.5151	0.4679	0.4681	0.5959	**0.6379**	0.6332
River	0.4542	0.4586	0.4151	0.4223	0.4267	0.4142	0.4246	0.4368	0.4249	0.4446	0.4642	0.4513	**0.4767**
Factory	0.3667	0.3776	0.3489	0.3767	0.3335	0.3591	0.3346	0.3583	0.3485	0.3567	0.3786	**0.3985**	0.3964
Tall building	0.4682	0.4775	0.4586	0.4868	0.4224	0.4346	0.4748	0.4765	0.4586	0.4581	0.4579	0.4733	**0.4973**
Residential	0.4586	0.4784	0.4274	0.4554	0.4275	0.4279	0.4756	0.4043	0.4252	0.4313	0.4476	**0.4897**	0.4856
Intersection	0.0946	0.1266	0.0889	**0.1157**	0.0947	0.0653	0.0668	0.0886	0.0646	0.0584	0.0536	0.0732	0.0619
Soccer field	0.4552	0.4345	0.4154	0.4268	0.4246	0.4275	0.4189	0.4252	0.4274	0.4269	0.4359	0.4617	**0.4752**
Bridge	0.6032	0.5774	0.5068	0.5256	0.5489	0.5275	0.5685	0.5246	0.5171	0.5375	0.5586	**0.6185**	0.6084
Park	0.5053	0.4559	0.4575	0.4643	0.4274	0.4585	0.4259	0.5142	0.4686	0.4579	0.4735	0.5014	**0.5165**
Farmland	0.1676	0.1635	0.1574	0.1542	0.1585	0.1474	0.1694	0.1775	0.1945	0.1981	0.2056	**0.2181**	0.2138
Townlet	0.4756	0.4582	0.4746	0.4477	0.4275	0.4038	0.4368	0.4227	0.4254	0.4262	0.4248	0.4457	**0.4795**
Racetrack	0.5595	0.5487	0.5191	0.5232	0.5576	0.5052	0.5335	0.5779	0.5419	0.5552	0.5575	0.5636	**0.5835**
Playground	0.2525	0.3046	0.2467	0.2192	0.2583	0.2542	0.2268	0.2349	0.2385	0.2324	0.2554	**0.3073**	0.2932
Wharf	0.3005	0.2881	0.2862	0.2784	0.2426	0.2543	0.2192	0.1956	0.2065	0.2143	0.2275	0.3016	**0.3026**
Valley	0.3559	0.3162	0.3624	0.3585	0.3542	0.3584	0.3436	**0.3784**	0.3568	0.3244	0.3351	0.3115	0.3144
Average	0.3840	0.3798	0.3631	0.3552	0.3557	0.3474	0.3470	0.3545	0.3456	0.3489	0.3671	0.4002	**0.4014**

The bold values represent the best performer.

TABLE IV
STANDARD ERROR OF F_1 SCORES OF DIFFERENT GOOGLE MAPS COMMUNITIES DISCOVERY ALGORITHMS

GM group	MMSB	BLDA	KC	HC	LC	CP	LRE	MAC	Gregory	Zhang	Yang	Ours(ICL)	Ours
Forest	0.0626	0.0534	0.0777	0.0559	0.0675	0.0461	0.0649	0.0462	0.0667	0.0556	0.0779	0.0519	0.0346
Aircraft	0.0565	0.0668	0.0779	0.0859	0.0681	0.0734	0.0562	0.0753	0.0642	0.0875	0.0568	0.0712	0.0437
Road	0.0635	0.0762	0.0645	0.0726	0.0674	0.0559	0.0774	0.0616	0.0642	0.0762	0.0688	0.0413	0.0421
Palace	0.0552	0.0651	0.0352	0.0689	0.0651	0.0559	0.0763	0.0713	0.0689	0.0624	0.0788	0.0507	0.0616
Sea	0.0723	0.0684	0.0834	0.0781	0.0742	0.0651	0.0589	0.0668	0.0624	0.0576	0.0734	0.0646	0.0692
Railway	0.0668	0.0543	0.0456	0.0585	0.0635	0.0751	0.0655	0.0536	0.0695	0.0661	0.0712	0.0732	0.0605
River	0.0686	0.0561	0.0661	0.0951	0.0559	0.0643	0.0675	0.0562	0.0741	0.0553	0.0628	0.0554	0.0419
Factory	0.0641	0.0652	0.0764	0.0603	0.0521	0.0691	0.0703	0.0624	0.0623	0.0561	0.0626	0.0731	0.0512
Tall building	0.0526	0.0653	0.0551	0.0679	0.0562	0.0691	0.0573	0.0645	0.0736	0.0564	0.0682	0.0423	0.0641
Residential	0.0643	0.0761	0.0568	0.0661	0.0657	0.0569	0.0748	0.0646	0.0569	0.0626	0.0542	0.0638	0.0425
Intersection	0.0662	0.0659	0.0635	0.0661	0.0578	0.0712	0.0674	0.0551	0.0586	0.0658	0.0443	0.0239	0.0517
Soccer field	0.0552	0.0661	0.0754	0.0563	0.0644	0.0431	0.0535	0.0664	0.0562	0.0489	0.0754	0.0661	0.0506
Bridge	0.0636	0.0549	0.0686	0.0743	0.0675	0.0556	0.0632	0.0689	0.0586	0.0782	0.0835	0.0724	0.0428
Park	0.0684	0.0582	0.0734	0.0679	0.0661	0.0668	0.0735	0.0696	0.0653	0.0732	0.0691	0.0589	0.0514
Farmland	0.0662	0.0551	0.0563	0.0667	0.0625	0.0461	0.0559	0.0651	0.0534	0.0619	0.0626	0.0576	0.0505
Townlet	0.0562	0.0418	0.0649	0.0525	0.0616	0.0526	0.0637	0.0893	0.0252	0.1056	0.0834	0.0725	0.0632
Racetrack	0.0612	0.0709	0.0675	0.0664	0.0789	0.0661	0.0464	0.0667	0.0569	0.0682	0.0756	0.0618	0.0505
Playground	0.0552	0.0661	0.0679	0.0686	0.0734	0.0856	0.0558	0.0345	0.0416	0.0745	0.0661	0.0515	0.0416
Wharf	0.0642	0.0595	0.0784	0.0742	0.0646	0.0524	0.0816	0.0761	0.0638	0.0555	0.0674	0.0543	0.0405
Valley	0.0634	0.0775	0.0562	0.0669	0.0756	0.0667	0.0645	0.0612	0.0675	0.0568	0.0626	0.0618	0.0406

of each Google Maps cluster is calculated by learning a GMM from the entire aerial photos. Next, inside different clusters, the well-known KL-divergence measure is deployed to calculate the difference among different distributions probabilistically learned. As can be seen from Table V, either the GMM, or the PLSA, or the LDA preforms worse than our designed probabilistic model. This is because that our proposed probabilistic model combines the superiorities of both the GMM and PLSA. Furthermore, by removing the regularization term, the communities discovery accuracy decreases sharply, particularly for Google Maps communities including users with few photos.

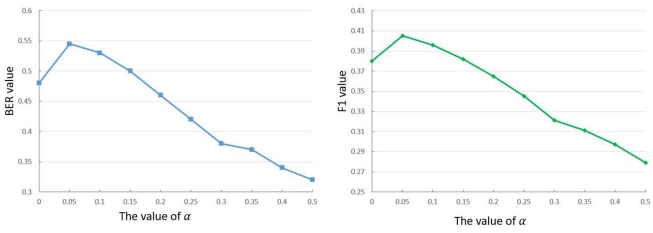

Fig. 10. (Left) BER and (right) F_1 values by varying α.

C. Parameter Analysis

As shown in Algorithm 1, we have multiple parameter sets that is tunable, *i.e.*,

1) the weight of the denoising term α;
2) the weight of the correlation preservation β;
3) the weight of the MF regularizer τ_1 and τ_2;
4) the weight of the regularizer in LTM.

To easily tune τ_1 and τ_2, we simply set $\tau_1 = \tau_2$.

In the first place, the BER and F_1 values is verified by changing α. This operation can determine the influences of denoising on Google Maps community labels. This parameter is tuned from 0 to 0.5 with a step of 0.05. We report the results in Fig. 10. For both BER and F_1 values, the highest accuracies are received when $\alpha = 0.05$. This result show that setting 0.05 to the denoising weight is an optimal choice.

TABLE V
KL-DIVERGENCES UNDER DIFFERENT TOPIC MODELS (GM MEANS GOOGLE MAPS)

GM group	GMM	CEM	PLSA	LDA	No Reg.	[24]	[25]	Ours
Forest	0.6546	0.7121	0.7435	0.7213	0.6867	0.7121	0.7301	**0.8213**
Aircraft	0.7021	0.7235	0.7453	0.7365	0.6754	0.6793	0.7110	**0.7340**
Road	0.7214	0.6934	0.7243	0.7136	0.6845	0.6910	0.8003	**0.8254**
Palace	0.8214	0.8325	0.8254	0.8009	0.8214	0.8442	0.8302	**0.8993**
Sea	0.7832	0.7634	0.7856	0.7931	0.7650	0.7731	0.7830	**0.8121**
Railway	0.7231	0.7436	0.7472	0.7658	0.7132	0.7449	0.7603	**0.8093**
River	0.8126	0.8043	0.8355	0.8360	0.7995	0.8112	0.8459	**0.9129**
Factory	0.8435	0.8324	0.8561	0.8245	0.8193	0.8243	0.8440	**0.8658**
Tall building	0.8325	0.8243	0.8547	0.8326	0.8216	0.8431	0.8563	**0.9154**
Residential	0.8768	0.8769	0.8343	0.8879	0.8547	0.8214	0.8317	**0.8657**
Intersection	0.7821	0.7768	0.8021	0.7962	0.7546	0.7658	0.7834	**0.8453**
Soccer field	0.7873	0.7547	0.8214	0.7887	0.7348	0.7432	0.7845	**0.8350**
Bridge	0.8074	0.8143	0.8254	0.8187	0.8435	0.8314	0.8034	**0.8436**
Park	0.8232	0.8156	0.8547	0.8657	0.8435	0.8324	0.8123	**0.8676**
Farmland	0.8657	0.8456	0.8657	0.8657	0.8343	0.8564	0.8845	**0.9103**
Townlet	0.8564	0.8657	0.8675	0.8436	0.8346	0.8554	0.8654	**0.9225**
Racetrack	0.8043	0.8165	0.8269	0.8047	0.8138	0.8343	0.8546	**0.8296**
Playground	0.8032	0.7854	0.8132	0.8184	0.8054	0.8121	0.8324	**0.8436**
Wharf	0.8437	0.8547	0.8216	0.8467	0.8369	0.8402	0.8554	**0.8657**
Valley	0.7768	0.7454	0.7984	0.7547	0.7456	0.7832	0.8002	**0.8165**

The bold values represent the best performer.

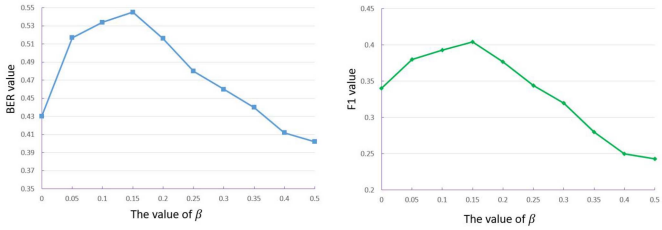

Fig. 11. (Left) BER and (right) F_1 values by varying β.

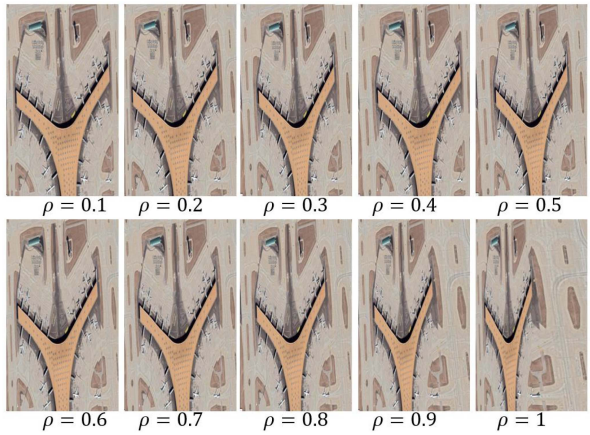

Fig. 12. Aerial photo retargeting results by tuning ρ.

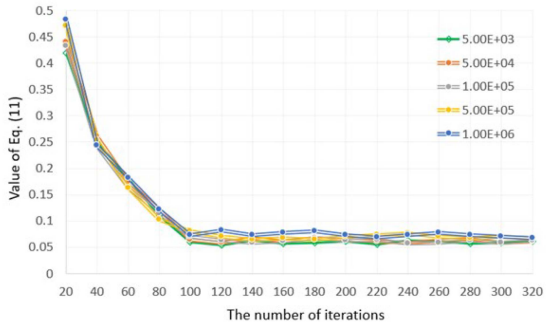

Fig. 13. Value of objective function by varying the iteration number.

Next, the accuracy of our Google Maps communities discovery is validated by adjusting β. This parameter indicates the importance of maintaining the intrinsic geometry among communities. Meanwhile, we adjust it from 0 to 0.5 with a step of 0.05. As displayed in Fig. 11, for both the BER and F_1, the calculated accuracy increases and subsequently peaks. Thereafter, the accuracy decreases to a relatively low value. This result shows that 0.05 is a good choice for β. At the same time, it verifies that on the massive-scale aerial photo set from Google Maps, communities' underlying relationships is equally treated as label denoising.

Third, we test our proposed method under various values of τ_1 and τ_2. This can regularize our designed enhanced MF framework. We adjust τ_1 and τ_2 from 0.01 to 0.1 with a step of 0.01. As can be seen from Fig. 11, the highest accuracy is obtained when $\tau_1 = \tau_2 = 0.15$. Last but not least, we report the retargeting performance by tuning ρ, the weight of regularizer of our proposed LTM. We notice that the optimal ρ is between 0.2 and 0.4 and depends on a specific aerial photo. As displayed in Fig. 12, for the test aerial photo. The most visually attractive retargeted aerial photo is produced when $\rho = 0.3$.

Actually, we cannot theoretically prove the convergence of the objective function as shown in (11). In this work, we experimentally evaluate the convergence of (11). More specifically, we use 5e3, 5e4, 1e5, 5e5, and 1e6 samples for learning the regularized LTM model. As shown in Fig. 13, the objective function converges fast when the iteration number increases from 1 to 100. When the iteration number exceeds 100, the objective function value becomes stable.

V. Conclusion

Aerial Photo retargeting is an indispensable algorithm in remote sensing. This article designed a novel community-based aerial photo retargeting algorithm by automatically discovering multiple Google Maps communities hidden in million-scale online aerial photos. A noise-tolerant deep MF model is proposed, which robustly mines communities reflecting different aerial photo styles. Afterward, a regularized latent topic model is presented to describe the style feature of aerial photos from a set of Google Maps communities. The overfitting issue can be effectively tackled. Lastly, a novel topic model is utilized to squeeze the test aerial photo by maximizing the posterior probability of the learned style distribution. Extensive experiments on 2 M aerial photos and the RetargetMe [33] have demonstrated the superiority of our approach.

One shortcoming of our method is the difficulty to determine the hidden community number. This is an open problem like deciding the cluster number for K-means. In our experiment, we tune the hidden community number until the best performance is observed. In the future, we plan to develop an algorithm to more intelligently determine the hidden community number.

References

[1] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann, "Link communities reveal multiscale complexity in networks," *Nature*, vol. 466, no. 7307, pp. 761–764, 2010.

[2] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing, "Mixed membership stochastic blockmodels," *J. Mach. Learn. Res.*, vol. 9, pp. 1981–2014, 2008.

[3] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," *ACM Trans. Graph.*, vol. 26, no. 3, pp. 21–32, 2007.

[4] R. Balasubramanyan and W. W. Cohen, "Block-LDA: Jointly modeling entity-annotated text and entity-entity links," *SDM*, 2011.

[5] C. Biernacki, G. Celeux, and G. Govaert, "Assessing a mixture model for clustering with the integrated completed likelihood," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 7, pp. 719–725, Jul. 2000.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.

[7] S. Castillo, T. Judd, and D. Gutierrez, "Using eye-tracking to assess different image retargeting methods," in *Proc. Symp. Appl. Percep. Graph. Visual.*, 2011, pp. 7–14.

[8] G. Celeux and G. Govaert, "Classification EM algorithm for clustering and two stochastic versions," *Comput. Statist. Data Anal.*, vol. 14, no. 2, pp. 315–332, 1993.

[9] Y.-W. Chen and C.-J. Lin, *Combining SVMs With Various Feature Selection Strategies*. Berlin, Germany: Springer, 2005.

[10] M. Frank, A. P. Streich, D. Basin, and J. M. Buhmann, "Multi-assignment clustering for Boolean data," *J. Mach. Learn. Res.*, vol. 13, pp. 459–489, 2012.

[11] B. Fuglede and F. Topsoe, "Jensen-Shannon divergence and hilbert space embedding, international symposium on information theory," 2004.

[12] S. Gregory, "A fast algorithm to find overlapping communities in networks," *Mach. Learn. Knowl. Discov. Databases*, vol. 5211, pp. 408–423, 2008.

[13] Y. Guo, F. Liu, J. Shi, Z. Zhou, and M. Gleicher, "Image retargeting using mesh parameterization," *IEEE Trans. Multimedia*, vol. 11, no. 5, pp. 856–867, Nov. 2009.

[14] K. He, G. Gkioxari, P. Dollr, and R. Girshick, "Mask R-CNN," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 3–11.

[15] T. Hofmann, "Probabilistic latent semantic analysis," in *Proc. Conf. Uncertainty Artif. Intell.*, 1999, pp. 44–51.

[16] R. Hong, L. Zhang, C. Zhang, and R. Zimmermann, "Flickr circles: Aesthetic tendency discovery by multi-view regularized topic modeling," *IEEE Trans. Multimedia*, vol. 18, no. 8, pp. 1555–1567, Aug. 2016.

[17] C. S. Johnson, "Hierarchical clustering schemes," *Psychometrika*, vol. 32, no. 3, pp. 241–254, 1967.

[18] Q. Ke and T. Kanade, "Robust L1 norm factorization in the presence of outliers and missing data by alternative convex programming," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 739–746.

[19] M. G. Kendall and B. B. Smith, "On the method of paired comparisons," *Biometrica*, vol. 31, pp. 324–345, 1940.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2012.

[21] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Phys. Rev. E*, vol. 78, 2008, Art. no. 046110.

[22] J. Leskovec, K. J. Lang, and M. W. Mahoney, "Empirical comparison of algorithms for network community detection," in *Proc. Web Conf.*, 2010, pp. 32–37.

[23] S. -S. Lin, I. -C. Yeh, C. -H. Lin, and T. -Y. Lee, "Patch-based image warping for content-aware retargeting," *IEEE Trans. Multimedia*, vol. 15, 2, pp. 411–419, Feb. 2013.

[24] Z. Duan, Y. Xu, B. Chen, D. Wang, and C. Wang, "TopicNet: Semantic graph-guided topic discovery," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2021, pp. 553–558.

[25] T. Nguyen and A. T. Luu, "Contrastive learning for neural topic model," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2021.

[26] W. Luo, X. Wang, and X. Tang, "Content-based photo quality assessment," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2206–2213.

[27] J. C. David MacKay, *Information Theory, Inference & Learning Algorithms*. New York, NY, USA: Cambridge Univ. Press, 2002.

[28] D. Martin and W. Powers, "Evaluation: From precision, recall and fmeasure to ROC, informedness, markedness and correlation," *J. Mach. Learn. Res.*, vol. 2, no. 1, pp. 37–63, 2011.

[29] G. Palla, I. Dernyi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, 2005.

[30] D. Panozzo, O. Weber, and O. Sorkine, "Robust image retargeting via axis-aligned deformation," *Comput. Graph. Forum*, vol. 31, no. 2, pp. 229–236, 2012.

[31] S. Papadimitriou, J. Sun, C. Faloutsos, and P. S. Yu, "Hierarchical, parameter-free community discovery," *Machine Learning and Knowledge Discovery in Databases*, vol. 5212. Berlin, Germany: Springer, 2008, pp. 170–187.

[32] Y. Pritch, E. Kav-Venaki, and S. Peleg, "Shift-map image editing," in *Proc. Int. Conf. Comput. Vis*, 2009, pp. 151–158.

[33] M. Rubinstein, D. Gutierrez, O. Sorkine, and A. Shamie, "A comparative study of image retargeting," *ACM Trans. Graph.*, vol. 29, no. 5, pp. 1–11, 2010.

[34] M. Rubinstein, A. Shamir, and S. Avidan, "Improved seam carving for video retargeting," *ACM Trans. Graph.*, vol. 27, no. 3, pp. 343–351, 2008.

[35] M. Rubinstein, A. Shamir, and S. Avidan, "Multi-operator media retargeting," *ACM Trans. Graph.*, vol. 28, no. 3, pp. 23–31, 2009.

[36] J. Sun and H. Ling, "Scale and object aware image thumbnailing," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 135–153, 2013.

[37] Y.-S. Wang, C.-L. Tai, O. Sorkine, and T.-Y. Lee, "Optimized scale-and-stretch for image resizing," *ACM Trans. Graph.*, vol. 27, no. 5, pp. 33–42, 2008.

[38] Y.-S. Wang, H.-C. Lin, O. Sorkine, and T.-Y. Lee, "Motion-based video retargeting with optimized crop-and-warp," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 637–648, 2010.

[39] L. Wolf and M. Guttmann, "Daniel cohen-or, non-homogeneous content-driven video retargeting," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2007, pp. 1–6.

[40] J. Yang and J. Leskovec, "Community affiliation graph model for overlapping network community detection," *Proc. Int. Conf. Des. Mater.*, 2012.

[41] T. Yang, R. Jin, Y. Chi, and S. Zhu, "Combining link and content for community detection: A discriminative approach," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2009.

[42] T. Yoshida, "Toward finding hidden communities based on user profile," in *Proc. Int. Conf. Data Mining Workshops*, 2010.

[43] L. Zhang, M. Song, Q. Zhao, X. Liu, J. Bu, and C. Chen, "Probabilistic graphlet transfer for photo cropping," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2887–2897, May 2013.

[44] Y. Zhang, J. Wang, Y. Wang, and L. Zhou, "Parallel community detection on large networks with propinquity dynamics," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2009, pp. 997–1006.