Generating Visual Story Graphs with Application to Photo Album Summarization

Bora Celikkale[a,*], Goksu Erdogan[a], Aykut Erdem[b,c], Erkut Erdem[a,c]

[a] *Hacettepe University Computer Vision Lab, Department of Computer Engineering, Hacettepe University, Ankara, Turkey*
[b] *Department of Computer Engineering, Koç University, İstanbul, Turkey*
[c] *KUIS AI Lab, Koç University, İstanbul, Turkey*

Abstract

Making sense of ever-growing amount of visual data available on the web is difficult, especially when considered in an unsupervised manner. As a step towards this goal, this study tackles a relatively less explored topic of generating structured summaries of large photo collections. Our framework relies on the notion of a story graph which captures the main narratives in the data and their relationships based on their visual, textual and spatio-temporal features. Its output is a directed graph with a set of possibly intersecting paths. Our proposed approach identifies coherent visual storylines and exploits sub-modularity to select a subset of these lines which covers the general narrative at most. Our experimental analysis reveals that extracted story graphs allow for obtaining better results when utilized as priors for photo album summarization. Moreover, our user studies show that our approach delivers better performance on next image prediction and coverage tasks than the state-of-the-art.

Keywords: `visual story graph`, `structured summarization`

1. Introduction

When we are planning a trip to a place we have never been before, we usually buy a guidebook or a travel app or visit websites such as `tripadvisor.com`

*Corresponding author
Email address: `boracelikkale@gmail.com` (Bora Celikkale)

or `wikitravel.org` to choose which places to visit and what to do in that destination. City guides which were prepared by professional travelers typically include essential information about the attractions, museums or parks in that city. Hence, each traveler, in a way, joins a collaborative act of living and enjoying the city and its culture. This joint act is clearly visible when we look at related travel photo albums shared on the web. Of course, the individual details can vary across trips, but common elements manifest themselves, providing collaborative stories about a city. Same landmark locations and attractions are visited regularly by tourists, and are being photographed again and again.

In this study, we propose a novel approach to automatically generate an informative visual summary of a specific city directly from a large set of travel photo albums related about that city. We formulate this task as a sub-modular optimization problem in which the structured summary is represented in terms of a *story graph*, providing information about different characteristics of a city. In general, a story graph allows to illustrate the common relationships between data samples in an informative manner, and has been a topic of interest in the scientific community lately. For instance, story graphs have been used to create summarizes of news articles [1], scientific papers [2], ego-centric videos [3] and the interactions among different characters in a movie or TV series [4].

Given tens of thousands of images of a city, in our work, we aim to identify a few storylines that (1) are coherent, i.e. each tells a coherent but different story, (2) cover most of the interesting attractions, i.e. they provide collective information regarding important and salient characteristics of the city, and (3) are connected, i.e. they effectively capture the hidden interconnections. Fig. 1 demonstrates an example story graph for the city of Istanbul, reconstructed automatically with our framework by analyzing lots of related travel photo albums. The main contributions of our work are as follows:

- We develop a collaborative summarization approach which exploits visual and textual data as well as geospatial and timestamp information to automatically extract a visual story graph for a large collection of photo

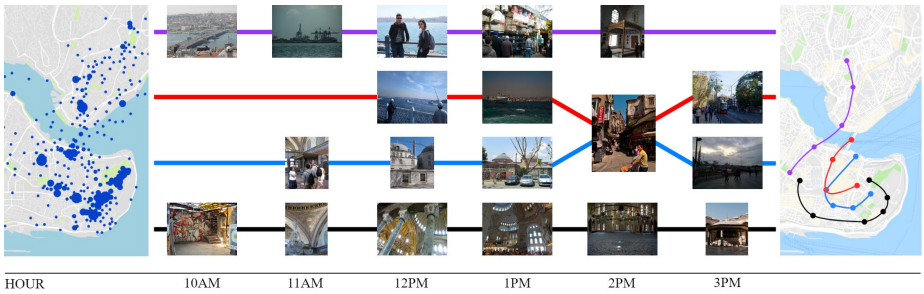


Figure 1: A story graph generated automatically by our approach for the city of Istanbul. On the left, we show the density map of the geo-tagged images collected from trips to the city of Istanbul. In the middle, we provide some sample storylines which cover coherent and distinct stories. On the right, we draw the story graph on the city map. For illustrative purposes, here we only show four storylines.

albums. Our formulation enforces maximum degrees of coherency, coverage and connectivity over the extracted storylines, and as it depends on sub-modularity, it is efficient and scalable.

- We introduce YFCC100M-CITIES dataset which includes images of six different cities, annotated with GPS, timestamp tags and textual keywords. It contains in total 132,346 images over 1566 photo albums from 323 users for 6 popular travel destinations in the world.

- We utilize the story graphs generated with our approach as structured abstractions of important concepts, landmarks and events within the photo collections, and demonstrate that they can be employed as a prior in photo album summarization to obtain state-of-the-art results.

- We further demonstrate the effectiveness of our framework with two user studies on next image prediction and tag coverage tasks. Our experimental results show that our model provides better results than the state-of-the-art.

Our code and data are publicly available at the project website: `hucvl.github.io/visual-storygraphs/`.

2. Related Work

We can group the related work on mining large photo collections into three different categories. First group of works focuses on data visualization aspect to allow a user to quickly explore large photo albums [5, 6, 7, 8, 9, 10, 11]. Second group addresses summarization of large photo collections by selecting a relatively small set of images based on some desired properties [12, 13, 3, 14, 15]. Last group of works, on the other hand, summarizes big visual data in a structured manner by means of story graphs [16, 4]. Instead of selecting a representative set of images, these works aim at reconstructing a narrative where each storyline in the graph reflects a major story arc in the image collections.

2.1. Exploratory Data Analysis and Visualization

Recently, there has been much interest in exploratory analysis of big visual data using visualization techniques. Platt *et al.* proposed a method to automatically create an overview of a collection based on clustering and then selecting the representative images from each cluster [5]. Cooper *et al.* suggested a similar framework that depends on clustering of photo collections based on similarities over appearance and temporal characteristics [6]. Kim *et al.* introduced a data-driven method to model and analyze the temporal evolution of the topics of the web images by constructing a large similarity graph of these images through a sequential Monte Carlo based method [8]. Berg and Berg developed an object-centric model to identify canonical images in a set of images collected for a specific object category [7]. Doersch *et al.* proposed a discriminative clustering approach to learn common and distinctive visual elements from large number of photos from a city [9]. Zhu *et al.* introduced a method which employs average images to let the users browse a large photo collection at ease [10]. More recently, Kleiman *et al.* suggested an approach to search, find and browse similar images on massively large image datasets by projecting their nearest neighbors in a high-dimensional feature space into a 2D layout [11].

2.2. Summarization of Visual Data

A large body of works aims at analyzing big visual data by selecting the most representative images among a given set of images by eliminating the redundant ones. The selection process amounts to capturing the most salient or interesting visual information depending on the task or motivation at hand. For instance, Simon *et al.* developed a photo collection summarization technique which extract the most interesting images over the collection by using a SIFT co-occurrences based clustering framework with a RANSAC loop [12]. Obrador *et al.* approached the summarization process from a supervised learning perspective in which the information from users' online social networks are used as cues [13]. Lu and Grauman proposed a summarization method for egocentric video which relies on segmenting the video into shots and identifying important objects in each shot and then extracts the summary by enforcing coherency based on common objects shared in consecutive shots [3]. Sadeghi *et al.* suggested a method for automatically creating a photo album from a large, unordered collection images, which can be also regarded as an unstructured summarization [14]. The authors, in particular, employ a discriminative structured model to capture compelling visual narratives through features encoding faces, scene context and certain visual attributes. More recently, Sigurdsson *et al.* have used recurrent neural networks to model long-term temporal relations among photo albums to extract visual storylines and summaries [15]. Similarly, Yu *et al.* [17] utilized recurrent neural networks to encode photos and select representative images that form a summary. They achieved improved results on summarization, story generation and album retrieval. Kim *et al.* [18] proposed an automatic summarization method that depends on scoring photos by their aesthetics, interestingness and memorability scores. Finally, Iyer *et al.* [19] developed an open-source library for video and album summarization covering data subset selection methods based on sub-modular functions. Specifically, this work allows for the use several summarization methods that employ either simple visual features such as color histograms or more complex semantic features extracted by deep neural networks.

2.3. Visual Story Graphs

Compared to the aforementioned groups of works, visual story graphs have been one of the least investigated topics in the computer vision literature. They serve as means for discovering hidden patterns and structures in large sets of images or videos while summarizing events and activities in the visual data. Xiong *et al.* [20] utilized story graphs to model egocentric videos in terms of story elements. They defined four sets of elements which correspond to actors, locations, supporting objects and events, respectively. Based on these elements, they proposed an inference algorithm to visualize the story on a timeline. In their pioneering work [16], Kim and Xing formulated generating visual story graphs as inferring a sparse time-varying directed graph from multiple photo albums which are collected on a single topic. Tapaswi *et al.* developed a similar graph based summary of videos over the interactions among different characters [4]. Like these studies, our approach also differs from the conventional summarization approaches in the sense that it outputs a structured summary depicting different aspects of the photo collections in the form of a story graph. In that regard, the most similar related work to our approach is the method of Kim and Xing [16]. However, our method is fundamentally different from this work in several aspects. Most notably, the approach in [16] does not explicitly try to maximize coverage and connectivity of the story graphs, whereas our proposed approach is built around these two fundamental concepts, together with the common notion of coherency. While the coverage leads to diversity of the images in the story graph, connectivity allows to extract the common aspects that are essential for photo album summarization. Moreover, our proposed framework employs a general formulation so that we can incorporate others modalities such as textual or GPS information to the story graph generation process. Lastly, while the story graphs in [16] are constructed with the nodes as the visual elements, the nodes of our story graphs correspond to individual images.

The story graphs generated from large photo collections can be also interpreted as a prior graph collaboratively constructed for a particular interest.

This property makes the proposed approach a convenient tool for photo album summarization since the generated story graphs both provide diverse information regarding the image collections but also encode particular aspects of the visual data that are shared among many users.

3. Approach

Here we introduce our approach to extract visual story graphs from image collections. We start with constructing dictionaries for visual and textual elements from the given sets of images. These elements serve as fundamental building blocks in finding coherent and intersecting storylines. In the rest of this section, we give the details of these steps, starting with a formal definition of story graphs.

3.1. Definition of a Story Graph

A story graph is a pair $\mathcal{S} = (G, \mathcal{P})$ where $G = (V, E)$ represents a directed graph, $\mathcal{P}$ denotes a set of chains (paths) which includes the storylines in G, the nodes of G correspond to the representative images from a large photo collection and its edges symbolize the connections among them. In an ideal case, a story graph, as a whole, should provide a visual collaborative summary of the photo collection from which it is extracted. This goal can be achieved by constructing it by considering three key properties, namely *coherence*, *coverage* and *connectivity* [21].

3.1.1. Coherence

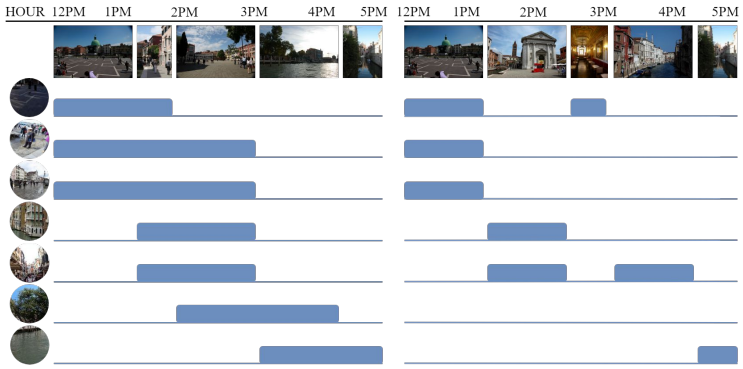
Intuitively, we want our story graphs to tell coherent stories. Hence, we need a mechanism to measure the consistency across each storyline of our story graph. We employ visual and textual elements as means for forming coherent visual stories through these storylines. Specifically, we connect the images with the visual and textual elements shared among them. We define the overall coherence gained by a storyline $\mathcal{P}_i = (p_1, ..., p_n) \in \mathcal{P}$ by the following equation:

$$Coherence(\mathcal{P}_i) = \min_{k=1..n-1} \sum_{e} \mathbb{1}(e \text{ is active in } p_k \text{ and } p_{k+1}) \quad (1)$$

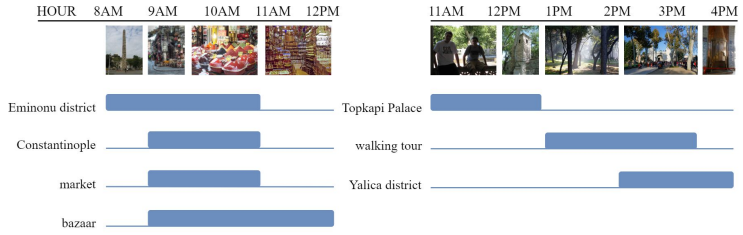
where $\{e\}$ denotes the set of elements, p_k represents the kth photo in the storyline. We consider an element e as active if its importance is above a certain pre-defined threshold for both p_k and p_{k+1}. We determined the threshold values by using grid search where we empirically sweep over a range of values and select the ones which lead to the best coherence and coverage scores. Accordingly, we set the threshold values as 0.22 for visual elements and 0.4 for textual elements.

In particular, here, we ensure that all the consecutive pairs of photos on the storyline share at least an element e which can either be a visual or a textual element. The function $\mathbb{1}$ is an indicator function which enforces that the element should be active among the photos p_k and p_{k+1}. The final coherence value is then determined by the weakest pair among the whole storyline. Hence, for a coherent chain, the behavior of all of the elements should provide a transition as smooth as possible throughout the storyline. Refer to 3.2.1 and Section 3.2.2 for the details of how we construct the visual and textual elements and decide whether an element is active or not for an image.

In Fig. 2, we show some sample coherent and incoherent chains based on visual and textual elements shared among the images in the chain and plots. As can be seen, the characteristics of the images change rapidly in an incoherent chain without producing consistent stories, which is valid for both visual and textual domains. In the figure, the active elements are demonstrated with blue bars spanning at least one photo among the chain. For the chains given on the left, the consecutive photos share some visual and textual elements. These elements give rise to more coherent transitions among the photos in the chain. On the other hand, most of the consecutive photos given on the right do not share any visual or textual element. Depending on our definition of coherence in Eqn. 1, this affects the consistency of the visual or the textual elements among the chains, creating incoherent stories.



(a)



(b)

Figure 2: Coherent and incoherent chain examples in terms of (a) visual elements and (b) textual elements. For each case, we show a number of images composing a story. The bars indicate the elements that are active on the images. The coherent chain given on the left tells a consistent story through smooth transitions over the active elements. On the other hand, within the incoherent chain shown on the right, the active elements change very rapidly over the images, which result in inconsistencies in the story told.

3.1.2. Coverage

Coverage property ensures that the photos among the storyline cover a diverse set of elements. That is, if a storyline sufficiently covers an element, there is no need to add it to the story graph. This brings the so-called diminishing return property that tells as new storylines are added to the graph, if the new storyline covers an element that has already been covered, it should contribute very little to the total coverage. With this property in mind, each element's

coverage through a story graph $\mathcal{S}$ is given by the following equation:

$$Coverage_{\mathcal{S}}(e) = 1 - \prod_{p \in photos(\mathcal{S})} (1 - Coverage_p(e)) \tag{2}$$

where $Coverage_p(e) \in [0, 1]$ denotes how important that element is for describing the photo p, defined differently for visual and textual elements. If the story graph $\mathcal{S}$ has photos covering the element e well, the coverage of the whole map on element e, $Coverage_S(e)$, will be close to 1 which means there is no need to select any other photos covering the same element e. If a new storyline has been added to the graph, it should cover different elements, resulting in a more diverse chains of photos. In our framework, visual elements connects photos via visual patches whereas textual ones creates semantic connections through textual keywords. However, of course, not all elements are equally important. Some visual elements such as the sky regions are so common among the images that it is not feasible to use them to form storylines. Similarly, specific textual keywords such as *White House* which shows a singular location should have higher importance than generic location names like *garden* or *museum*.

Total coverage of a story graph is then computed as the summation of the coverage of both visual and textual elements as given below:

$$Coverage(\mathcal{S}) = \alpha \sum_{v \in \mathcal{V}} Coverage_{\mathcal{S}}(v) + (1 - \alpha) \sum_{t \in \mathcal{T}} Coverage_{\mathcal{S}}(t) \tag{3}$$

where $v \in \mathcal{V}$ denotes a visual element, $t \in \mathcal{T}$ represents a textual element, and $\alpha \in (0, 1)$ is a scalar representing relative significance of textual and visual elements. In our experiments, we empirically set the value of α to 0.1.

3.1.3. Connectivity

Connectivity enforces that the storylines should share some photos which amounts to the crossing points between the chains. This is a unique property that gives a story graph a nonlinear story structure as compared to the simple linear story model. The story graph is more informative when it shows hidden connections between diverse paths. In other words, without connectivity, the output will be linear summaries of individual photo collections. Although it

seems contradicting with the coverage property, we look for minor connections between storylines after selecting a diverse set, preserving diversity together with a few individual photo similarities. Formally, the connectivity of a graph can be defined in terms of a value denoting the sum of the number of lines that intersect in story graph $\mathcal{S}$:

$$Connectivity(\mathcal{S}) = \sum_{i<j} \mathbb{1}(\mathcal{P}_i \cap \mathcal{P}_j \neq \emptyset) \tag{4}$$

with $\mathcal{P}_i$ and $\mathcal{P}_j$ denoting the ith and jth storylines in the story graph $\mathcal{S}$.

3.2. Constructing the Story Graph

We cast the story graph construction as an optimization task defined over extracted coherent storylines $\mathcal{S} = (\mathcal{P}_i, .., \mathcal{P}_n)$. That is, we compute the optimal story graph $\mathcal{S}^*$ by first extracting most coherent storylines and then selecting a diverse set of important ones which intersect with each other to a certain extent by considering the following equation that is built upon *Coverage*, and *Connectivity* characteristics:

$$\begin{aligned} \mathcal{S}^* \quad = \quad & \text{argmax}_{\mathcal{S}}\ Connectivity(\mathcal{S}) \\ & \text{s.t. } Coverage(\mathcal{S}) \geq C \end{aligned} \tag{5}$$

where C denotes a coverage score that is smaller than the highest coverage score that can be obtained without considering the connectivity property. Note that $Coherency(\mathcal{S})$ does not appear here as we already start with the coherent storylines extracted from the constructed coherence graph. Moreover, $Coverage(\mathcal{S})$ appears as a constraint since we first try to maximize the coverage score and then maximize the connectivity accordingly in the final stage, as explained below.

An optimum approach to find $\mathcal{S}^*$ is not trivial, hence, instead, we use a greedy approach by exploiting the sub-modular structure that exist in our problem. That is, we first maximize coverage and then try to maximize connectivity over storylines by allowing some decrease in the maximum possible coverage score (please refer to Section 3.2.4 for the details about how the maximal coverage can be defined. The whole algorithm is summarized in Algorithm 1.

Algorithm 1 Steps of finding the optimal story graph $\mathcal{S}^*$ from a large collection of images denoted by $\mathcal{I}$

1: **for** each image p_i in the input photo collection $\mathcal{I}$ **do**
2: Estimate the importance weights for the visual elements (Section 3.2.1)
3: Estimate the importance weights for the textual elements (Section 3.2.2)
4: Compute the coherence graph G based on the transitions over elements (Section 3.2.3)
5: Extract a set of high coverage chains from G (Section 3.2.4)
6: Perform a local search to improve the connectivity (Section 3.2.5)

3.2.1. Visual representation

Our visual representations are based on bag of visual elements. In particular, we approach the extraction of the visual elements from a dictionary learning perspective. In particular, we employ a recently proposed deep feature called Regional Maximum Activation of Convolutions (RMAC) [22] which achieves the state-of-the-art performance for the image retrieval task. Specifically, the RMAC representation that we use in our work depends on the VGG-16 [23] model pre-trained on ImageNet. It is extracted from the last pooling layer, resulting in a 3D tensor having $7 \times 7 \times 512$ dimensions where the scalar 512 denotes the number of filters. Then, from these response maps they sample R uniform square regions at L different scales with 40% overlap. In our case, we set $L = 4$ in order to get finer scale visual elements. It is important to note that increasing the value of L might allow us to capture more finer scale visual elements, albeit with an increase in computational cost. For each region $r \in R$ max-pooling is performed on each channel and obtained a feature vector of 512-dimensions as shown in Eqn. 6. The last step is the L_2-normalization to get a single region vector.

$$f_{r \in R} = [f_{r,1} \dots f_{r,i} \dots f_{r,K}]^T \tag{6}$$

In our work, we cluster these region features with K-Means clustering algorithm and form the visual dictionary for a city accordingly. We set the size

Figure 3: Sample visual elements from the visual dictionary constructed from the Paris vacation photo albums. These elements are visualized by finding the image patches having the closest RMAC representations [22]. While some of them captures the details from touristic attractions (left), some correspond to very ordinary regions such as trees, clouds, and sky (right).

of this dictionary as 1024. This approach captures various structures that persistently exist in the image collections, reflecting the visual characteristics of a city and the popular landmarks within. In Fig. 3, we demonstrate sample image regions which are close to some of the visual elements from the Paris dataset. As can be seen, some of these regions correspond to the details from the touristic attractions such as *Eiffel Tower, Arc de Triomphe, Notre Dame and Louvre Museum* as given on the left. However, since our dictionary learning procedure does not use any prior knowledge about the cities, some of the extracted visual elements might correspond to very common image regions such as sky, trees, etc. as shown on the right. Hence, for each image p_i we assign a certain importance weight to each visual element v, which is defined inversely proportional to the number of occurrences of this visual element in the whole image collection.

Each image is decomposed into a set of local image regions, each encoded via a RMAC feature. Then, Locality-constrained Linear Coding (LLC) [24] is applied over these regions to obtain the final representation by max pooling of each region's code vector over the extracted visual elements. LLC encoding yields a sparse representation where only the most prominent visual elements are considered in the final representation. Importances of visual elements ($Coverage_{\mathcal{S}}(v)$) are then defined in terms of this LLC encoding scheme.

In Eqn. 1, the coherence score is estimated through the active visual elements over a storyline. The decision about whether a visual element is active or not is made by inspecting the weights of this visual element within the LLC encodings of the image pairs. If they are above a certain threshold, we assume the element is shared between the images and considered as active.

3.2.2. Textual Representation

In our work, we represent the images in the photo-collections in a multi-modal manner. As we mentioned earlier, representing images visually is carried out by first learning a visual dictionary from the training images and then by extracting visual elements from each image. Apart from this, we also consider a semantic representation of images that depends on textual information. In particular, each image can be tagged by a list of words by employing a pre-trained set of image classifiers that identify the visual characteristics of the image. In our work, we alternatively assume that each image has been already associated with a set of keywords. By this way, we can utilize a dictionary of words extracted from all of the images in the collections and then represent each image in terms of these keywords. To determine the importance of textual elements, we employ a tf-idf weighting scheme.

Similar to the visual elements, the coherence score due from the textual elements is computed by taking into account the textual elements that are shared over a storyline. While deciding these shared textual elements, we utilize their importance scores ($Coverage_{\mathcal{S}}(t)$) as indicated by their tf-idf weights. We assume that a textual element is active if its score is above a pre-defined threshold value.

3.2.3. Finding Coherent Storylines

We start with modeling storylines by the transitions of the extracted visual and textual elements. The brute-force solution to optimize the energy function in Eqn. 5 inspects every pair of images for the occurrence of all elements, and thus it takes time proportional to $N^2 \times D$ where N is the number of images

and D is the total number of elements. Since this is intractable for large image collections, we use a divide-and-conquer approach to build storylines. First, we extract short chains of images with smooth transitions being observed over some visual and/or textual elements. Then, we combine these short chains which overlap through some common images to obtain longer storylines that constitute our coherence graph G.

Our algorithm starts with a RANSAC [25] loop where at each iteration we randomly choose two images from the collection, which share at least a visual or textual element to satisfy the coherence property and which correspond to the end points of a short chain. Hence, to determine the images in between these two, for each shared element we search for images that also share the same element. Specifically, we enforce a smooth transition across the storyline as in [16]. For each shared visual element of the end point images, we fit a line over the activation scores coming from the LLC encoding [24] and validate the consistency of a candidate image by analyzing how well it fits to this linear activation transition function [21] by its corresponding element. For each shared textual element of the end point images, we check whether the element is active in the candidate image or not.

In our framework, we also utilize additional meta-data about the photos, namely the time-stamps and GPS location information to enforce additional constraints to improve the quality of the transitions. First, each image over a storyline should be captured after the time the photo preceding it is taken. This eliminates the possibility of ambiguous ordering of images such that a night time image follows a day time. Second, an image should be close to its previous image in geospatial terms. This enhances both the structure and the overall visual appearance of the storyline in that nearby locations are more likely to share similar visual structures. In our experiments, we empirically set the length of the short chains as 3. Larger values, in general, fail to find sufficient number of high quality chains.

Once we extract the coherent short chains, the next step is to construct a coherence graph G. We combine the shorter chains by the common images

that they share and accordingly obtain longer chains, each of which denotes a coherent storyline.

3.2.4. Finding Storylines with High Coverage

In the previous subsection, we show how to extract all coherent storylines on a coherence graph G we build based on short chains. Finding storylines with high coverage corresponds to selecting a subset of those from G that maximize the coverage as Eqn. 3 indicates. This can be formulated as an *orienteering problem*, aka *prize-collecting TSP* [26, 21], in which the goal is to maximize rewards collected while walking on the graph subject to a budget on the tour length and given two endpoints. The reward function is given by $f : 2^V \rightarrow \mathbb{R}^+$, which returns a non-negative value to every subset of nodes. Exhaustively searching for an optimum solution is infeasible but we can exploit the submodularity of our coverage function (Eqn. 2) where greedy algorithms with good approximation guarantees exist in the literature [26].

A set function $f : 2^V \rightarrow \mathcal{R}$ is *submodular* if $f(\mathcal{A} \cup a) - f(\mathcal{A}) \geq f(\mathcal{B} \cup a) - f(\mathcal{B})$ and for all $\mathcal{A} \subseteq \mathcal{B} \subseteq V$. This property is referred to as the *diminishing returns*, meaning that adding a new item to a smaller set provides a larger gain than adding it to a larger set.

After we extract our coherent storylines, we define the following incremental coverage notion to measure the gain in the coverage score when we add the storyline to our story graph S for each storyline $\mathcal{P}_i$ as follows:

$$IncCoverage(\mathcal{P}_i|\mathcal{S}) = Coverage(\mathcal{P}_i \cup \mathcal{S}) - Coverage(\mathcal{S}) \tag{7}$$

To sum up, in order to find the set of storylines that have the highest coverage over the visual elements, we follow an incremental search strategy. Starting with the storyline having the highest coverage value, we gradually enlarge the story graph by analyze each not included storyline by its contribution to the current coverage (Eqn. 7) and add the one that contributes the most. This procedure is repeated until there is no additional gain.

3.2.5. Increasing Connectivity

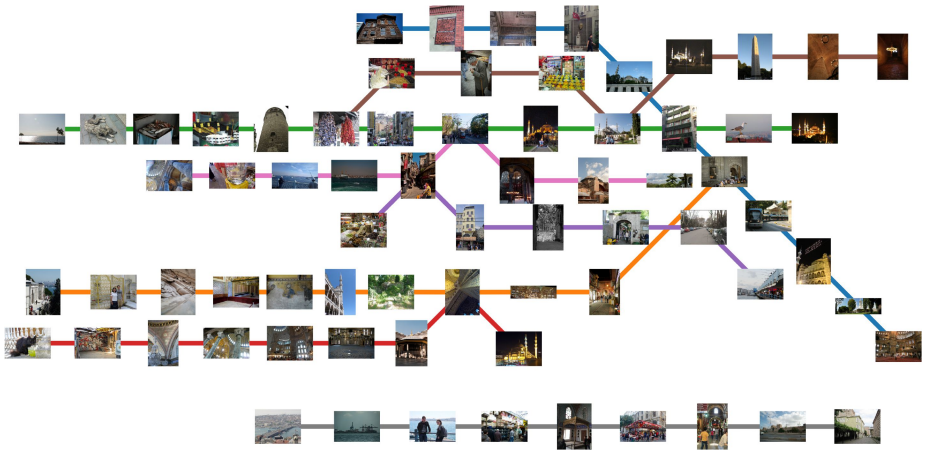
Increasing the connectivity is important to discover nonlinear story structures. We perform a local search operation on the extracted coherence graph G by using the story graph $\mathcal{S}^+$ with the highest coverage as an initial point. In particular, we fix the storyline having the highest individual coverage and perform a search among all of the other storylines forming the coherence graph G. Our aim is to find storylines alternative to the ones in $\mathcal{S}^+$, which increases the connectivity by allowing a reasonable amount of degradation in the total coverage value. Of course, the key question here is how much coverage drop can be tolerated. Allowing too much drop in the coverage results in story graphs with low coverage whereas limiting it to a low value prevents finding an appropriate chains for the replacements. In our work, we empirically observe that a 7% drop in the total coverage score generally gives satisfactory results. In Fig. 4, we provide the story graphs for the cities of Istanbul and Paris, which are automatically constructed by our approach from large sets of travel photo albums collected from the web.

3.3. Story-Graph Guided Photo Album Summarization

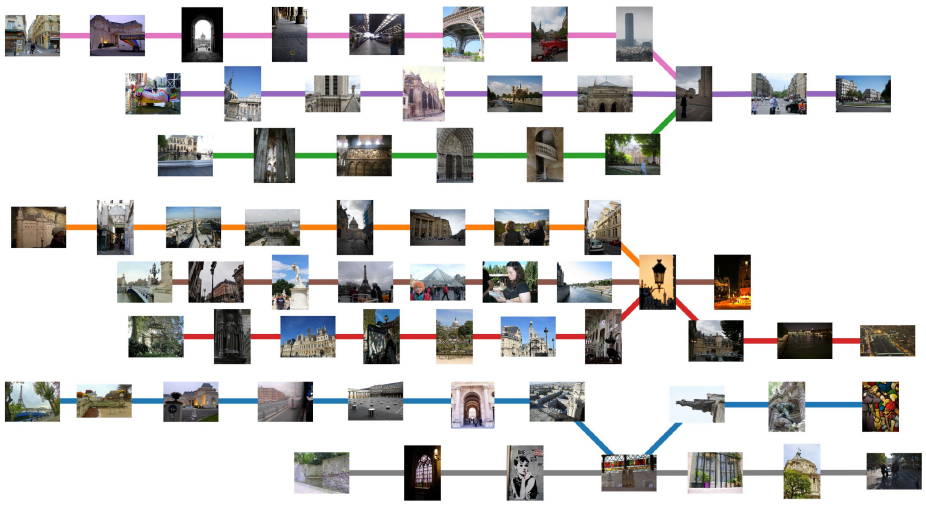
In the previous sections, we develop a method to generate visual story graphs from a large collection for photo albums. These story graphs are collaborative structured summaries containing coherent visual storylines and providing a comprehensive overview of specific topics of interest. With these characteristics, story graphs can be interpreted as prior graphs representing important concepts, landmarks and events within the photo collections.

In that regard, in this section, we demonstrate a way to obtain more effective summaries of photo albums and albums that cover the topics encoded in the story graphs generated by our approach.

Given a photo album $\mathbb{X}$, our goal is to extract a small number of images from $\mathbb{X}$ that represents the whole set. We additionally assume that another set of images are given in the form of a story graph $\mathbb{Y}$. Here, we formulate the summarization task as a subset selection task. For this purpose, we par-



(a)



(b)

Figure 4: The story graphs of (a) Istanbul and (b) Paris, which are based on travel photo albums collected from the web. The nodes (images) of the graphs are arranged based on the available timestamp information.

ticularly employ the DS3 algorithm [27] which formulates subset selection as a row-sparsity regularized trace minimization problem which can be easily solved via convex optimization.

In short, the DS3 algorithm solves a special subset selection problem when side information is available in the form of dissimilarities between the source set $\mathbb{X}$ and a target set $\mathbb{Y}$, defined as:

$$\begin{aligned} \min_{\{z_{ij}\}} \quad & \lambda \sum_{i=1}^{M} \|\mathbf{z}_i\|_p + \sum_{j=1}^{N} \sum_{i=1}^{M} d_{ij} z_{ij} \\ \text{s.t.} \quad & \sum_{i=1}^{M} z_{ij} = 1,\ \forall j;\ z_{ij} \geq 0,\ \forall i, j \end{aligned} \tag{8}$$

where z_{ij} is the indicator of the source item $x_i \in \mathbb{X}$ representing the target item $y_j \in \mathbb{Y}$ and d_{ij} denotes the dissimilarity between x_i and y_j. In our experiments, we use the KL-divergence as our dissimilarity measure. The parameter λ provides a trade-off between the number of representative samples and the encoding quality with smaller values of λ causing more number of samples selected as representative. Here, the first term penalizes the size of the representative subset and the second term is the encoding cost. In [27], the authors show that an optimal solution can be found using an Alternating Direction Method of Multipliers (ADMM) approach in an effective manner.

Notice that here we suggest to let $\mathbb{Y}$ denote the set of images available in the input story graph. Hence, while extracting a summary from the given photo album denoted by $\mathbb{X}$, the representative samples of $\mathbb{X}$ in the generated summary cover the themes available in $\mathbb{Y}$. Alternatively, we can let $\mathbb{Y} = \mathbb{X}$ by selecting the target set same as the source set. If this is the case, it becomes a self-summarization problem [27].

4. YFCC100M-CITIES Dataset

To evaluate our proposed approach for story graph generation, we need a large scale multi-modal dataset containing several different photo albums about a city. As far as we know, no large-scale dataset, which consists of images with textual tags, timestamps and GPS information, is freely available in the literature. Hence, we curated a new dataset by selecting and annotating images from the publicly available YFCC100M dataset [28]. In short, YFCC100M dataset [28] contains 99.2M photos and associated metadata such

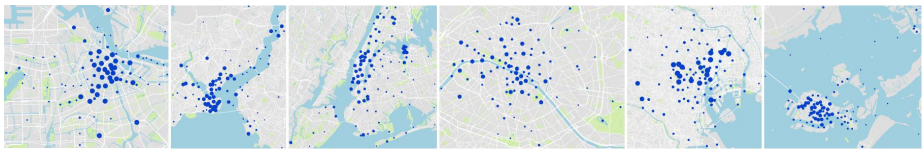


Figure 5: The distribution of photos in our YFCC100M-CITIES dataset. The area of a circle is proportional to the density of the photos in that location.

as time stamps, geolocation information and keywords from Flickr. However, most of the time, the user generated keywords are noisy, and since the users are from different countries, they use different languages while providing them, and thus YFCC100M dataset cannot directly be used within a multimodal summarization task in its current form.

In our work, we particularly collected vacation photographs from 6 different cities, namely Amsterdam, Istanbul, New York, Paris, Tokyo, Venice which are among the most visited cities around the world. We eliminated the photo albums that consist of only close-up pictures of humans or cover just one topic such as flowers in a garden. For user generated keywords, we filtered out highly generic words or words that are unrelated to the topic of interest. We then grouped similar and synonym words into common concepts by taking into account non-English words as well. In total, we have collected 132K geotagged images from 323 users and 1.5K photo albums. Fig. 5 and Table 1 show the basic statistics of our dataset, which we named as YFCC100M-CITIES dataset.

5. Experiments

An extensive application that make use of a story graph is photo album summarization task. We performed an extrinsic evaluation of our story graphs in which we leverage them as a prior to guide photo album summarization (Section 5.1). Another common approach to the visual evaluation task is performing user studies. Based on our formalism, a good story graph must first

Table 1: Statistics of YFCC100M-CITIES.

City	**Number of**	**Number of**	**Number of**	**Number of**
Albums	**Users**	**Photo Sets**	**Photos**	**Unique Words**
Amsterdam	39	100	9,923	1,460
Istanbul	58	167	13,645	979
New York	54	428	30,443	18,538
Paris	39	178	21,819	1,521
Tokyo	71	514	36,787	4,007
Venice	62	179	19,729	2,032
Total	323	1,566	132,346	25,118

meet two criteria. It must be composed of *coherent* chains and these chains should all together should *cover* most of the important aspects. However, it is difficult to quantitatively evaluate these two notions so we decided to perform controlled user studies, on which we compare against the previous work by Kim and Xing [16]. To assess coherence, we employ the next image prediction task proposed in [16] (Section 5.2), but to evaluate coverage we devised a new experiment (Section 5.3) since there has been no particular attention to this essential property.

5.1. Photo Album Summarization

As mentioned earlier, the story graphs provide a collaborative summary of photo albums on specific themes, which can be used as priors. As for our first experiment, we conduct an extrinsic evaluation of our proposed summarization framework by utilizing story graphs as priors in photo album summarization. For this task, we collected 6 additional photo albums from Flickr. These photo albums are related to trips to the either one of the six cities in our data set but they do not exist in YFCC100M dataset. Each one of these photo albums consists of 100 different images. Moreover, each album is annotated with 20 different human-generated summaries, which are obtained using a web tool via crowdsourcing. In particular, for every album we show the users the whole set of

photos from the album, which are sorted by time-stamp, and then let them select 10 most representative images among all these images. In our experimental evaluation, we employ these human-generated summaries as the ground truth.

For comparison, we test two simple baselines that are based on uniform sampling (Uniform) and K-Means clustering (K-Means), the skipping recurrent neural network model (S-RNN) by Sigurdsson *et al.* [15], two subset selection based summarization methods by Iyer *et al.* [19], which respectively employ simple color histograms of hue and saturation channels (DSS-S), and deep features from the last fully connected layer of the VGG-16 network (DSS-D)[1] and the DS3 model performing self-summarization with $\mathbb{Y} = \mathbb{X}$. In addition to those, we constructed three story graphs using our framework by taking into account (1) only visual features ($\mathbb{S}^V$), (2) visual features along with GPS information ($\mathbb{S}^{VG}$), and (3) both visual, GPS and textual information ($\mathbb{S}^{VGT}$). Lastly, to observe how different network architectures affect the quality of the representations, we conducted additional summarization experiments by using the visual features extracted from ResNet-50 [29]. In particular, we employ ResNet-50 as the backbone architecture for RMAC instead of the VGG-16 network. ResNet-50 based features, however, require more physical space as compared to those based on VGG-16 as their dimension is $4\times$ larger. Hence, due to the memory constraints, we could not reconstruct the story graphs with ResNet-50 features, but we encode the images in both the story graphs and the photo albums via a ResNet-50 based visual dictionary. We basically carried out the self summarization experiments ($\mathbb{Y} = \mathbb{X}$) and the summarization experiments with our best model ($\mathbb{Y} = \mathbb{S}^{VGT}$) by additionaly using these alternative visual dictionaries.

We quantitatively evaluate the performance with two metrics: F-measure [30] and V-ROUGE [31]. F-measure measures the accuracy of automatic summaries considering both precision and recall with respect to human-generated summaries. V-ROUGE is an extension of the ROUGE metric used for document

[1]Here, we intentionally use VGG-16 model for a fair comparison with our approach, which employs features from the same base network.

summarization and measures how the automatic summaries correlate with the human-generated ones based on occurrence-counts of visual elements.

Fig. 6 shows some sample summarization results for the Istanbul trip album. Uniform baseline gives a low quality summary in that it includes similar and semantically uninteresting images. K-means baseline generates a summary that lacks a coherent story considering the content of the input photo album. The summaries obtained with these approaches contain repetitive structures such as multiple photos of mosque ceilings and patterns. Even for the S-RNN method, which is a recently proposed summarization approach, we observe such an unwanted behavior. The results obtained with the DSS methods are much better in terms of diversity but they do not demonstrate a smooth transition between the selected images, hence the coherency is low. The DS3 model in general selects more complex and diverse photos as the notion of diversity is explicitly built in the selection mechanism. Especially, the variants that use story graphs as a prior performs much better as the summaries posses continuity of events. Among those, $\mathbb{S}^{VGT}$ seems to provide the best result as the images selected for the summary cover the main events depicted in the input photo album, and they appear to be semantically more close the summary by a human. Overall, both of our qualitative and quantitative results show that photo album summarization can benefit from exploiting visual story graphs as a prior to encourage producing more coherent summaries.

In Table 2 and Table 3, we report the V-ROUGE and F-measure scores, respectively. As can be seen, the quality of summaries obtained with the simple baselines, Uniform and K-means, is lower than the other approaches. S-RNN also gives unsatisfactory results although its formulation is based on modeling how a story evolves within a photo album. DSS method with simple features (DSS-S) produces slightly better summaries than S-RNN, but it is beaten by DSS-D, which is somewhat expected as deep features provide better semantic representations. The summaries obtained by different versions of our proposed framework, $\mathbb{S}^{V}$, $\mathbb{S}^{VG}$ and $\mathbb{S}^{VGT}$, are far better than the competing methods, including the deep approaches deep learning based models S-RNN [15] and DSS-

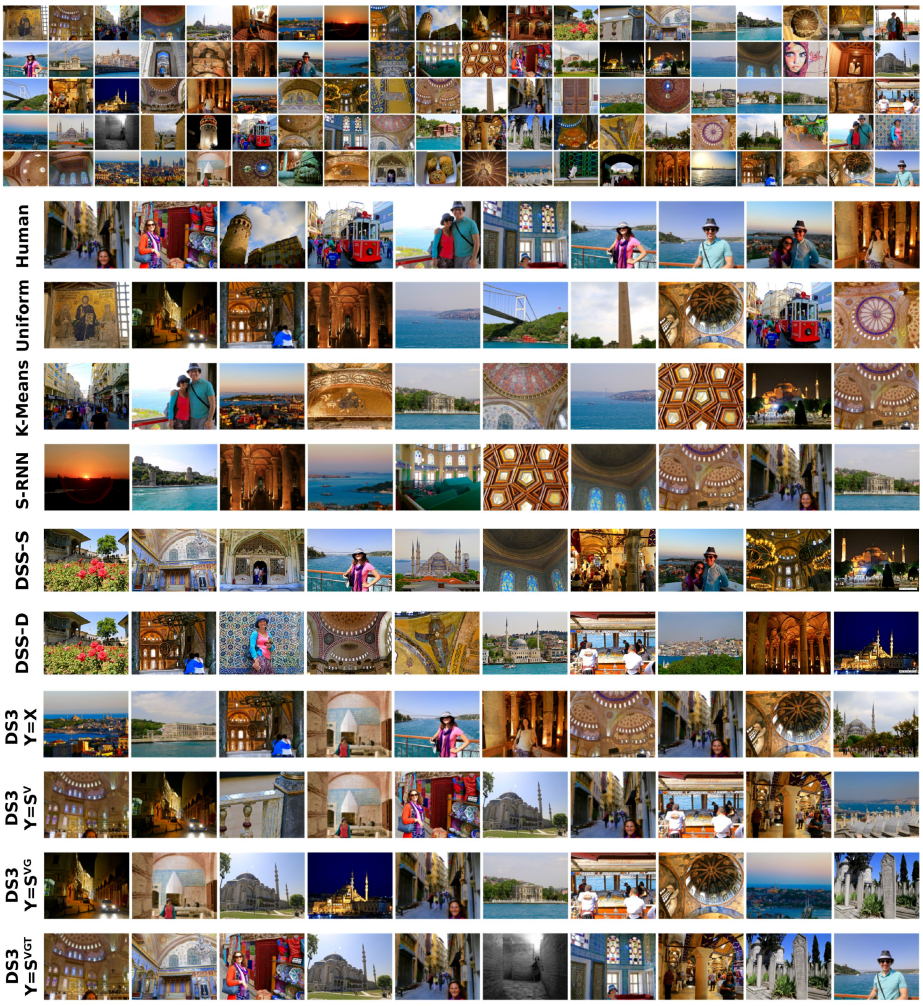


Figure 6: Sample summarization results. Top: Input photo album. Bottom: Visual summaries done by a human, the baselines approaches Uniform Sampling, K-Means clustering, S-RNN [15], DSS-S [19] and DSS-D [19] along with the ones obtained via the DS3 method using self summarization ($\mathbb{Y} = \mathbb{X}$), the story graphs constructed with visual features ($\mathbb{Y} = \mathbb{S}^V$), both visual and GPS features ($\mathbb{Y} = \mathbb{S}^{VG}$) and all visual, GPS and textual features ($\mathbb{Y} = \mathbb{S}^{VGT}$).

D [19]. Moreover, we observe that our fully featured story graph $\mathbb{S}^{VGT}$ which employs both visual, GPS and textual information, in general, achieves the best summarization performance. When we examine the cases where we switch from

Table 2: V-ROUGE scores for the summarization experiments.

Photo Album	**Amsterdam**	**Istanbul**	**New York**	**Paris**	**Tokyo**	**Venice**
	Trip	**Trip**	**Trip**	**Trip**	**Trip**	**Trip**
Uniform	0.31	0.38	0.48	0.33	0.45	0.45
K-means	0.45	0.26	0.39	0.37	0.39	0.29
S-RNN	0.30	0.39	0.41	0.35	0.42	0.33
DSS-S	0.38	0.41	0.39	0.38	0.39	0.24
DSS-D	0.40	0.44	0.49	0.39	0.52	0.27
DS3 ($\mathbb{Y} = \mathbb{X}$)	0.48	0.47	0.56	0.52	0.49	0.54
DS3 ($\mathbb{Y} = \mathbb{S}^{V}$)	0.48	**0.53**	0.61	0.44	0.52	0.57
DS3 ($\mathbb{Y} = \mathbb{S}^{VG}$)	0.46	0.42	0.50	0.47	0.53	0.58
DS3 ($\mathbb{Y} = \mathbb{S}^{VGT}$)	0.56	0.49	0.67	0.56	**0.63**	**0.66**
DS3 ($\mathbb{Y} = \mathbb{X}$)$^{\text{ResNet-50}}$	0.56	0.48	0.61	0.52	0.57	0.53
DS3 ($\mathbb{Y} = \mathbb{S}^{VGT}$)$^{\text{ResNet-50}}$	**0.59**	0.52	**0.70**	**0.69**	0.55	0.50

VGG-16 to ResNet-50 as the backbone for our visual features, for some cities, we see a slight increase in the V-ROUGE scores. However, the F-measure scores show no solid improvements. This indicates that ResNet-50 does not provide a significant gain in terms of the summarization performance. We believe that the structure of the prior (i.e. the story graphs) plays a more important role than the visual representations when selecting the most representative photos for the visual summaries.

5.2. Next Image Prediction

In our second experiment, we focus on the next image prediction task suggested in [16], which captures a story graph's ability in predicting what happens next given an input image. This task is related to evaluating coherence aspect of story graphs as the purpose is to identify how related the output image is to the query in terms of spatio-temporal continuity. We first select a small subset of canonical images for each city by simply clustering the entire set of photos into 50 clusters and retrieving the most photos that are close to the cluster centers. Given a query image, we localize the most similar photo in the reconstructed

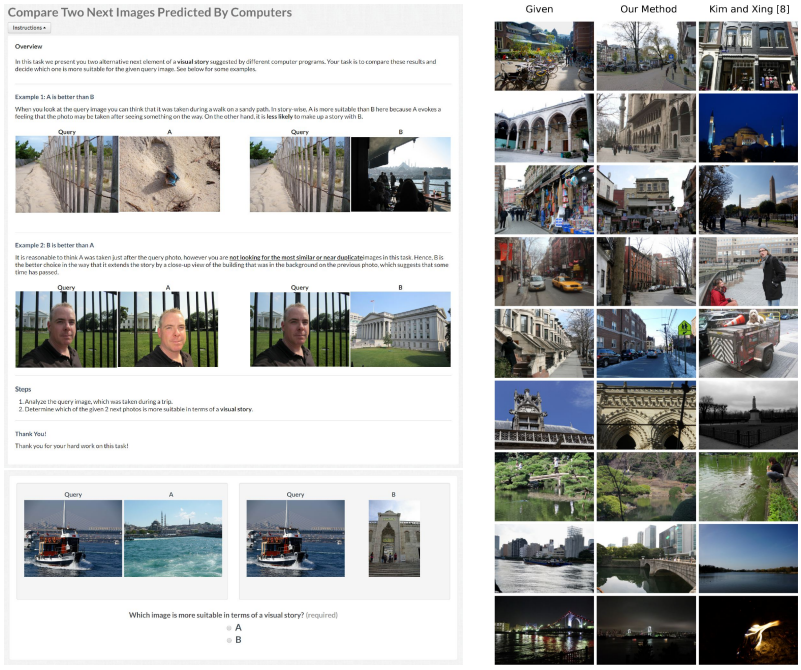
Table 3: F-measure scores for the summarization experiments.

Photo Album	**Amsterdam**	**Istanbul**	**New York**	**Paris**	**Tokyo**	**Venice**
	Trip	**Trip**	**Trip**	**Trip**	**Trip**	**Trip**
Uniform	0.02	0.10	0.17	0.05	0.11	0.13
K-means	0.12	0.05	0.06	0.09	0.12	0.05
S-RNN	0.05	0.10	0.11	0.07	0.08	0.08
DSS-S	0.07	0.07	0.09	0.16	0.08	0.16
DSS-D	0.12	0.11	0.17	0.15	0.13	0.18
DS3 ($\mathbb{Y} = \mathbb{X}$)	0.16	0.08	0.12	0.13	0.08	0.20
DS3 ($\mathbb{Y} = \mathbb{S}^{V}$)	**0.25**	0.10	0.14	0.17	0.04	0.17
DS3 ($\mathbb{Y} = \mathbb{S}^{VG}$)	0.15	**0.12**	0.16	0.19	**0.15**	0.21
DS3 ($\mathbb{Y} = \mathbb{S}^{VGT}$)	0.14	0.10	**0.19**	**0.21**	0.11	**0.23**
DS3 ($\mathbb{Y} = \mathbb{X}$)$^{\text{ResNet-50}}$	0.20	0.08	0.07	0.12	0.13	0.15
DS3 ($\mathbb{Y} = \mathbb{S}^{VGT}$)$^{\text{ResNet-50}}$	0.09	0.11	0.12	0.18	0.12	0.15

story graph and retrieve its next image in the corresponding chain. In the user study, subjects are presented with results obtained with our approach and with those by Kim and Xing's method [16] and are asked to choose the one which is the most likely sequence (Fig. 7(a)). We perform the user study on Figure Eight platform[2] in which a total of 331 workers have participated. For each test question, we obtain responses from at least 10 users. Fig. 7(b) shows examples of the next likely images predicted by our approach and the competing method. The results of the pairwise preference tests are given in Table 4. On average, our predictions are favored 61% of the time. One noticeable observation found here is that our approach outperforms the method of Kim and Xing's method by a large margin for the city of New York. We think that this can be explained by the statistics given in Table 1. The New York dataset has exceedingly large number of unique words obtained from the user provided tags. As our definition of the coherence property is based on shared visual and textual elements, our

[2]Figure Eight is a web-based data annotation company which can be accessed from https://www.figure-eight.com

framework is expected to get more coherent photo chains which correspond to story lines from high number of unique words.



(a) (b)

Figure 7: Next image prediction. (a) Screenshot of the user interface used in our experiments on the next image prediction task. (b) Example images predicted by our algorithm and the method of Kim and Xing [16].

Table 4: User study results for the next image prediction task. The preference rate denotes the percentage of comparisons in which the users favor one method over the other. On average, our predictions are preferred 61% of the time against the state-of-the-art method in [16].

	Amsterdam	Istanbul	New York	Paris	Tokyo	Venice	Average
Kim and Xing [16]	43.1	48.6	12.3	45.3	42.4	44.9	39.4
Ours ($\mathbb{S}^{VGT}$)	**56.9**	**51.4**	**87.7**	**54.7**	**57.6**	**55.1**	**60.6**

5.3. Coverage

In our last set of experiments, we compare the coverage of the story graphs generated by our approach and the method of Kim and Xing [16]. For each city, we first identified a diverse set of tags about the points of interest and attractions in that city via inspecting the user tags from YFCC100M dataset and additionally using the Google search engine. Table 5 shows these tags. For each tag we also provide an illustrative image just to give the workers an opinion about what that tag is about. In the user study, we then show the photos compiled from the reconstructed story graphs and ask users to select the tags that they think are relevant to one or more images displayed to them (Fig. 8). For each tag we estimate the percentage of workers who selected the tag for that particular story graph. Then, we calculate the average selection rate through all the tags to get the final coverage rate of the story graph with respect to all the tags of that city. We perform the user study on Figure Eight platform in which a total of 238 workers have participated. For each test question, we obtain responses from at least 10 users. For each city, our story graph achieves a higher coverage rate than that of Kim and Xing [16]. On average, our proposed approach covers 46.3% of the tags whereas the method of Kim and Xing covers 34.8% (Table 6). This demonstrates that the photos in the story graphs extracted by our method include points of interests and more interesting locations for a city, resulting in a more inclusive and covering visual narrative of a city.

6. Conclusion

In this study, we proposed an approach to automatically extract story graphs from large collections of photo albums, which serves as a collaborative and structured summary of these albums. We treated this task as a submodular optimization problem and formulated a greedy approach to find a graph that maximizes the degrees of coherence, coverage and connectivity of the storylines. We demonstrate that story graphs obtained with our approach can be utilized for photo album summarization. In particular, our story graphs can

Table 5: Tags used in coverage experiments.

City	**Tags**
Amsterdam	*Anne Frank House, Canals, Church, Cycling, Dam Square, Fine arts, Food, NEMO Science Museum, Night life, Parks, Port of Amsterdam, Rijksmuseum, Royal Palace Amsterdam, Van Gogh Museum, Windmills*
Istanbul	*Basilica Cistern, Bath houses, Beyoglu Street, Bosphorus Bridge, City Walls, Galata Tower, Grand Bazaar, Maiden's Tower, Mosques, Museums, Obelisk of Theodosius, Palace, Sea tour, Turkish food*
New York	*Broadway, Brooklyn Bridge, Cathedral, Chinatown, Coney Island, Grand Terminal, Museums, NYC Subway, Parks, Public Library, Skyscrappers, Statue of Liberty, Times Square, Wall Street*
Paris	*Arc De Triomphe, Art, Cafes, Champs Élysées, Eiffel Tower, Fountains, Louvre Museum, Montmartre, Moulin Rouge, Musée d'Orsay, Notre-Dame de Paris, Pantheon, Parks and gardens, Versailles*
Tokyo	*Disneyland, Edo-Tokyo, Fish Market, Ginza Crossing, Japanese food, Kabuki Theatre, Mount Fuji, Museums, Parks, Rainbow Bridge, Roppongi, Sanrio Puroland, Skytree, Subway and trains, Temples, Tokyo Imperial Palace, Traditional clothes*
Venice	*Bridge of Sighs, Carnival Masks, Fine Arts, Glassworks, Gondola, Grand Canal, Venetian Lagoon, Lido, Museums, Palazzo Ducale, Rialto, San Marco, St Mark's Campanile, Venetian Churches*

Table 6: User study results for the coverage task. The scores denote the average percentage of the tags selected by the workers for images included in the story graphs. On average, our story graphs cover 46% of the tags, providing a significantly higher rate than that of the state-of-the-art method in [16].

	Amsterdam	Istanbul	New York	Paris	Tokyo	Venice	Average
Kim and Xing [16]	34.7	24.3	30.0	41.9	26.7	50.9	34.8
Ours ($\mathbb{S}^{VGT}$)	**45.3**	**50.1**	**38.6**	**43.0**	**43.4**	**57.1**	**46.3**

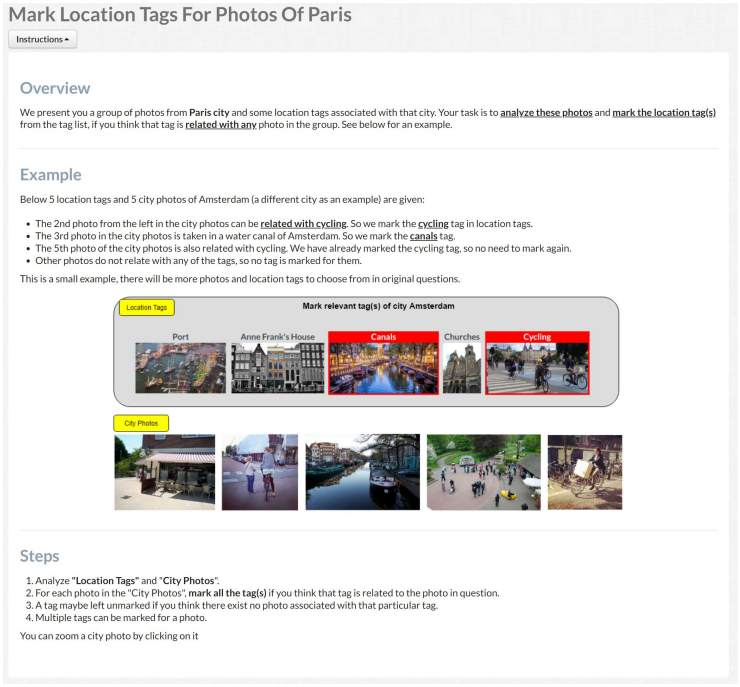


Figure 8: A screenshot of the user interface used in our experiments on the coverage task.

be interpreted as a kind of prior that represent important concepts, landmarks and events depicted in the large photo collections, and hence, the images in the story graphs can serve as a measure of representativeness while extracting summary of a photo album of similar theme. Our experimental analysis reveals that the story graphs obtained by our approach allow to obtain better performances than the previous approaches for three different tasks including photo album summarization, next image prediction, tag coverage. For future work, it would be interesting to investigate how intrinsic image properties such as interestingness or aesthetics affect the extracted story graphs. Moreover, we plan to include some kind of personalization to allow the users to enforce some preferences while constructing the story graphs.

Acknowledgement

This work was supported in part by TUBA GEBIP fellowship awarded to E. Erdem and by The Scientific and Technological Research Council of Turkey (TUBITAK) Primary Subjects R&D Funding Program Award No. 116E685.

References

[1] D. Shahaf, C. Guestrin, Connecting the dots between news articles, in: ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), 2010, pp. 623–632.

[2] D. Shahaf, C. Guestrin, E. Horvitz, Metro maps of science, in: ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), 2012, pp. 1122–1130.

[3] Z. Lu, K. Grauman, Story-driven summarization for egocentric video, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 2714–2721.

[4] M. Tapaswi, M. Bauml, R. Stiefelhagen, Storygraphs: visualizing character interactions as a timeline, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 827–834.

[5] J. C. Platt, M. Czerwinski, B. Field, et al., PhotoTOC: Automatic clustering for browsing personal photographs, in: IEEE Joint International Conference on Information, Communications and Signal Processing (ICICS) and Pacific Rim Conference on Multimedia (PCM), 2003, pp. 6–10.

[6] M. Cooper, J. Foote, A. Girgensohn, L. Wilcox, Temporal event clustering for digital photo collections, ACM Transactions on Multimedia Computing, Communications, and Applications 1 (3) (2005) 269–288.

[7] T. L. Berg, A. C. Berg, Finding iconic images, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2009, pp. 1–8.

[8] G. Kim, E. P. Xing, A. Torralba, Modeling and analysis of dynamic behaviors of web image collections, in: European Conference on Computer Vision (ECCV), 2010, pp. 85–98.

[9] C. Doersch, S. Singh, A. Gupta, J. Sivic, A. Efros, What makes Paris look like Paris?, ACM Transactions on Graphics 31 (4).

[10] J.-Y. Zhu, Y. J. Lee, A. A. Efros, AverageExplorer: Interactive exploration and alignment of visual data collections, ACM Transactions on Graphics 33 (4) (2014) 160.

[11] Y. Kleiman, J. Lanir, D. Danon, Y. Felberbaum, D. Cohen-Or, Dynamicmaps: Similarity-based browsing through a massive set of images, in: ACM Conference on Human Factors in Computing Systems (CHI), 2015, pp. 995–1004.

[12] I. Simon, N. Snavely, S. M. Seitz, Scene summarization for online image collections, in: IEEE International Conference on Computer Vision (ICCV), 2007, pp. 1–8.

[13] P. Obrador, R. De Oliveira, N. Oliver, Supporting personal photo storytelling for social albums, in: ACM International Conference on Multimedia (ACM MM), ACM, 2010, pp. 561–570.

[14] F. Sadeghi, J. R. Tena, A. Farhadi, L. Sigal, Learning to select and order vacation photographs, in: IEEE Winter Conference on Applications of Computer Vision (WACV), 2015, pp. 510–517.

[15] G. A. Sigurdsson, X. Chen, A. Gupta, Learning visual storylines with skipping recurrent neural networks, in: European Conference on Computer Vision (ECCV), 2016, pp. 71–88.

[16] G. Kim, E. P. Xing, Reconstructing storyline graphs for image recommendation from web community photos, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 3882–3889.

[17] L. Yu, M. Bansal, T. L. Berg, Hierarchically-attentive rnn for album summarization and storytelling, arXiv preprint arXiv:1708.02977.

[18] J.-H. Kim, J.-S. Lee, Travel photo album summarization based on aesthetic quality, interestingness, and memorableness, in: 2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), IEEE, 2016, pp. 1–5.

[19] R. Iyer, P. Dubal, K. Dargan, S. Kothawade, R. Mahadev, V. Kaushal, Visdss: An open-source toolkit for visual data selection and summarization, arXiv preprint arXiv:1809.08846.

[20] B. Xiong, G. Kim, L. Sigal, Storyline representation of egocentric videos with an applications to story-based search, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 4525–4533.

[21] D. Shahaf, C. Guestrin, E. Horvitz, Trains of thought: Generating information maps, in: ACM International Conference on World Wide Web (WWW), 2012, pp. 899–908.

[22] G. Tolias, R. Sicre, H. Jégou, Particular object retrieval with integral max-pooling of cnn activations, in: International Conference on Learning Representations (ICLR), 2016.

[23] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: International Conference on Learning Representations (ICLR), 2014.

[24] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, Locality-constrained linear coding for image classification, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 3360–3367.

[25] M. A. Fischler, R. C. Bolles, Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, Communications of the ACM 24 (6) (1981) 381–395.

[26] C. Chekuri, M. Pal, A recursive greedy algorithm for walks in directed graphs, in: IEEE Symposium on Foundations of Computer Science (FOCS), 2005.

[27] E. Elhamifar, G. Sapiro, S. S. Sastry, Dissimilarity-based sparse subset selection, IEEE Transactions on Pattern Analysis and Machine Intelligence 38 (11) (2016) 2182–2197.

[28] B. Thomee, D. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, L. Li, YFCC100M: The new data in multimedia research, Communications of the ACM 52 (2) (2016) 64–73.

[29] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[30] N. Chinchor, Muc-4 evaluation metrics, in: Proceedings of the 4th conference on Message understanding, Association for Computational Linguistics, 1992, pp. 22–29.

[31] S. Tschiatschek, R. K. Iyer, H. Wei, J. A. Bilmes, Learning mixtures of submodular functions for image collection summarization, in: Advances in Neural Information Processing Systems (NIPS), 2014, pp. 1413–1421.