Using Text to Teach Image Retrieval

Haoyu Dong
Duke University
haoyu.dong151@duke.edu

Ze Wang Qiang Qiu
Purdue University
{wang5026, qqiu}@purdue.edu

Guillermo Sapiro
Duke University
guillermo.sapiro@duke.edu

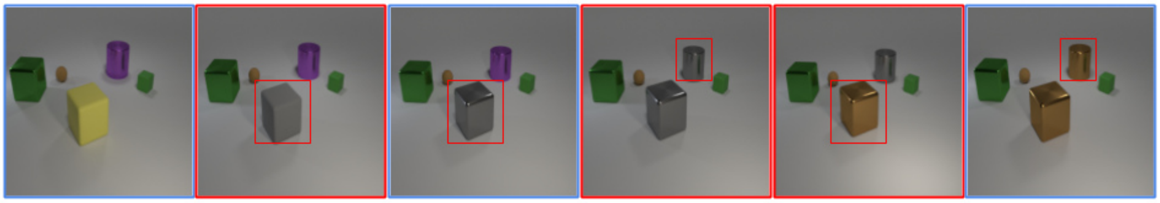

Figure 1: Visualization of a smooth geodesic path as obtained with the framework proposed in this paper. The images are from our newly introduced dataset, CCI. Each image presents a node in a graph, and its boundary indicates the source of the node (blue for image domain, and red for text domain). The modification between adjacent images is marked with a red square, corresponding to a single attribute change per step on the joint embedding geodesic path. Best viewed in color.

Abstract

Image retrieval relies heavily on the quality of the data modeling and the distance measurement in the feature space. Building on the concept of image manifold, we first propose to represent the feature space of images, learned via neural networks, as a graph. Neighborhoods in the feature space are now defined by the geodesic distance between images, represented as graph vertices or manifold samples. When limited images are available, this manifold is sparsely sampled, making the geodesic computation and the corresponding retrieval harder. To address this, we augment the manifold samples with geometrically aligned text, thereby using a plethora of sentences to teach us about images. In addition to extensive results on standard datasets illustrating the power of text to help in image retrieval, a new public dataset based on CLEVR is introduced to quantify the semantic similarity between visual data and text data. The experimental results show that the joint embedding manifold is a robust representation, allowing it to be a better basis to perform image retrieval given only an image and a textual instruction on the desired modifications over the image.

1. Introduction

Retrieval is the task of finding the most relevant object in a database given a query. Recent works have grown the interest in cross-domain retrieval, especially between image and text domains. The image-text retrieval task can be generally summarized into two directions. One is to match the corresponding images given sentence queries, or vice versa; this has been one of the most popular branches in the field of cross-domain research [8, 32]. The other direction, *e.g.*, [12, 30], is to conduct text-based retrieval; the task uses an image with a textual instruction describing some desired modifications to the image as a query, and the target image is the modified image. In both scenarios, the search is done by mapping the queries and database objects to a joint feature space.

Although remarkable progress has been achieved, the basic frameworks of retrieval are mostly built upon the assumption that the similarity of images is well approximated by either negative Euclidean distance or negative cosine distance, both assuming that the features are in an Euclidean space. This can be sub-optimal under the assumption that images reside on a low-dimensional manifold within a high-dimensional feature space [3], where a geodesic distance can better define the relationship between objects. Furthermore, since we usually only have access to a limited number of samples in the visual domain, the feature space is under sampled, and thus sparse [19]. A sparse feature space means that some points can be far away from all the other points, making it hard to define proper neighborhoods for them. This is considered as the sparsity problem, and we address this issue as well.

In this work, we model the manifold of image features, learned via neural networks, as a graph, where each vertex represents an image. Considering that a manifold is only locally homeomorphic to Euclidean space, we build an edge between a pair of vertices only when their Euclidean distance is small, as standard in the point clouds literature [29]. Since edges represent distances, our image graph is weighted, and we compute the geodesic distance as the sum of weights along the shortest path instead of just as the number of edges. To evaluate the effectiveness of the geodesic distance measurement, we study a label retrieval task which aims at classifying every node in a graph with labels only available for a small subset. The motivation behind this task is from semi-supervised learning, where the goal is to propagate label information in a naturally defined fashion. This learning task has a key assumption, that points in the same locality are likely to share the same label [6, 38]. Therefore, it can be seen as a natural way to measure the robustness of the feature representation, *i.e.*, images with the same label are closer than ones with different labels.

The geodesic distance and manifold concept further allow us to consider the sparsity problem, meaning the limited number of samples (vertices), in the retrieval task. When the number of samples is limited, some samples are far from the rest. Then the degree of these samples are small, meaning they only have a few or even do not have geodesic neighbors. We say a point is **retrievable** if it has a geodesic neighbor in the small subset that contains label information, and otherwise it is **unretrievable**. We propose to exploit plethora of text in order to learn a visual-semantic embedding space to reduce the number of unretrievable points and to improve the geodesic computation accuracy. The objective of a joint embedding space is thus to improve the learned image manifold representation by adding, to the original image samples (vertices), new manifold samples via semantically related text. These newly added text samples can interpolate the visual feature space, and thus increase the number of geodesic neighbors for each point.

The geodesic path in a graph can be seen as a series of modifications from the starting image to the ending one. We consider a path to be "smooth" if the difference between any two adjacent vertices along the path is small and interpretable, *i.e.*, we can use a sentence to describe this difference. Then we can study how well the textual domain is incorporated into the visual domain by examining the increase in the number of "smooth" geodesic paths when texts are added to construct the graph. To quantify the concept of smoothness, we use the CLEVR framework [17] to build a new dataset, CLEVR-Change-Iter (CCI). The framework renders simple 3D images; each image contains multiple objects, and each object is determined by several attributes (colors, shapes, materials, etc.). We then define the difference between two images to be small and interpretable *iff* they differ by only one attribute of an object (see Sec. 4.2 for a formal definition). An example of a "smooth" path is shown in Fig. 1. CCI is constructed in a way that there exists at least one "smooth" path between any two points.

We conduct the label retrieval tasks on two natural image sets: ADE20K [37] and OpenImage [21]. We show that a geodesic neighbor leads to a better retrieval performance than an Euclidean neighbor does. Furthermore, we observe that using image and textual information together to represent the manifold allows a more completed neighborhood description, where more points are retrievable. To validate the "smooth" path counting task, we build a cross-modal embedding space on CCI. We compare our learned text features with random text features to show that merely having more samples does not increase as many "smooth" paths as semantically similar features do.

Interestingly, we also find that the manifold in the joint feature space, with only image samples, outperforms the one in a pretrained image feature space in the label retrieval task. This means that we can use text as privilege information to learn a more robust representation of images. To validate this statement, we use the collection of image features from the joint space as a basis for text-based image retrieval. We obverse consistent improvements over different embedding methods and different datasets. We also show that our new dataset, CCI, can be used for the text-based retrieval task.

In conclusion, our contribution is three-fold.

- We demonstrate that the geodesic distance is a more accurate measurement to the relationships between objects. Adding corresponding texts can alleviate the sparsity problem, and improve the embedding manifold representation.
- We show that corresponding texts can also be used as privileged information for text-based retrieval.
- We introduce a new public dataset, CCI, along with a new criteria to evaluate the quality of the aligment between the textual domain and the visual domain. The dataset can be use for text-based retrieval task as well.

2. Prior Work

Fusion of Vision and Language. There is a growing interest to solve the problems at the intersection of computer vision and natural language processing; the problems range from transferring information from one domain to another, *i.e.*, image captioning [1, 9, 34] and text-to-image generation [15, 24], to integrating information from both domains to solve questions that require cross-modal knowledge, such as visual question answering (VQA) [2, 18, 28] and novel object captioning [16, 22, 35]. These cross-domain problems usually require a task-specific framework to answer, although some works try to learn a visual-semantic joint embedding space that can be more adaptive to multiple tasks [8, 32]. Of particular interest, we find that cross-domain learning brings alignment that can be viewed as an unsupervised way to solve problems. For example, [13] finds that regions which fire high in spatial activation maps correspond to the relevant objects described in the speech; [27] builds an unsupervised translation system with the assumption that sentences in different languages appearing with visually similar frames are correlated. In both cases, specific information, *i.e.*, segments of objects and parallel corpora, is not available. In our work, we find that captions (text) can be seen as extra samples from the same distribution (manifold) to provide a richer representation of the image space helping in label retrieval, even though the label for each sentence is unknown.

Cross Domain Retrieval. Deep learning based image retrieval usually finds images with similar content by Convolutional Neural Networks (CNNs) that are pretrained on classification problems, and has gained significant progress [10, 31]. Cross-domain retrieval extends the problem by considering non-image queries, such as text to image retrieval [33] and sketch to image retrieval [25]. The works [12, 30] further study text-based retrieval, where each query is composed of an image plus some instructions describing the desired modifications to that image; the framework allows users to provide feedback to product search. Based on our observation that image samples are more robustly represented in the joint space, and the task's tendency to incorporate modification sentences into an image representation, we use the image features from the joint space as a basis to perform the text-based retrieval task. Parallel to our work, [7] also has incorporated side information into the text-based retrieval task. Though the approach is similar, we see the improvement as a proof of concept.

Semi-Supervised Learning. Semi-supervised learning has two key assumptions of consistency: (1) nearby points are likely to share the same label; and (2) points with similar structure are likely to have the same label [38]. Recent works that use graph convolutional networks (GCN) follow the second assumption and achieve promising results [20]. Although we also represent image features as a graph, the graph is used to calculate shortest path distances and to propagate label information. Our approach is still K-Nearest Neighbor (KNN) based, which follows the first assumption. We thus believe the improvement in label retrieval performance reflects that the image representation is more robust.

3. Method

In this section, we first describe the proposed method of learning a visual-semantic joint embedding space by incorporating text information into the visual domain. Then we present the approach to construct a graph in this joint space. Finally, we show how to use image features from the visual-semantic domain as a basis to conduct text-based retrieval. Figure 2 illustrates the proposed architecture.

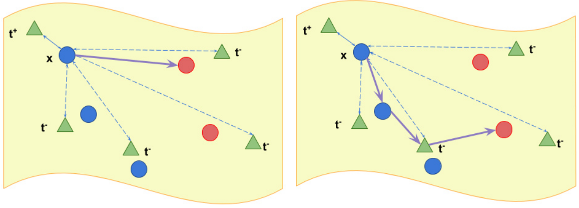

Figure 2: Illustration of learning the cross-domain feature space and two corresponding distances. An image point (blue circle) is encouraged to be close to its corresponding text point (green triangle) t^+, and far from non corresponding text points t^-. Left and right figures show different nearest points are found in the target set (red circles) by ranking Euclidean and geodesic distances, respectively. Best viewed in color.

3.1. Cross-Modal Embedding

To learn the manifold and correspondence between images and texts in a joint feature space, we use a two-branch neural network similar to [32, 36]. In the following, X and Y will be the collection of images and texts respectively, and $x \in X$ and $y \in Y$ will be the individual image and text.

Visual Embedding Module. We use a CNN to project x to image features. In our experiments, we use ResNet18 [14] and replace the classification layer with a fully-connected (FC) layer with d units.

Textual Embedding Module. We encode y using Universal Sentence Encoder (USE) [5], which encodes the inputs at the sentence level. The encoded features are passed through two FC layers with Rectified Linear Unit (ReLU) activation, where the second FC has d units.

Semantic Projection Layers. The projection layer for each module is a fully-connected layer, followed by L2 normalization, that maps the features (image or text) onto a joint latent space. Denote ψ_i as the joint embedding for x_i and ϕ_i as the joint embedding for y_i, the objective of a joint space is that matched image and text features should be close to each other, and unmatched ones should be far away. More precisely, consider we have a training mini-batch of B samples, $\{\psi_i, \phi_i\}_{i=1}^{B}$. For each ψ_i, the positive example is ϕ_i, and we consider all other samples in the mini-batch as negative samples. This leads to the following cross entropy loss:

$$L_{joint} = -\sum_{i=1}^{B} log\{\frac{exp\{\kappa(\psi_i, \phi_i)\}}{\sum_{j=1}^{B} exp\{\kappa(\psi_i, \phi_j)\}}\}, \qquad (1)$$

where κ is the dot product, which is equivalent to the negative Euclidean distance after the features are normalized.

We acknowledge that this assumption penalizes all $y_{j,j\neq i}$, but some y_j can be relevant to x_i if their corresponding images x_j are visually similar to x_i. However, we empirically find that taking this into consideration, *i.e.*, by removing samples with high dot products from the negative set, decreases the performance.

3.2. Manifold in the Joint Space

In order to promote better feature correspondence between image features and text features, we perform feature alignments to reduce the distance between pairs. Specifically, we adopt the iterative point alignment (ICP) algorithm [4]. Since the pair information is known, the rotation T has a closed form solution and thus we only run the algorithm once. Given the features of the collection of images and texts, and applying the ICP transformation on the image features, $(T(\Psi), \Phi)$, we can now construct the graph G that represents the manifold formed by the images. Each node (vertex) is either an image embedding or a text embedding, and edges encode distances between two features. To better mimic traversing on the manifold, we encode the distances as great circle distances on the unit sphere. Following the local-Euclidean property of a manifold, an edge (i, j) exists *iff* the distance between i and j is lower than a given threshold, which is an hyperparameter to define locality and usually set to satisfy that $O(|E|) = O(|V|)$.

3.2.1 Manifold Evaluation

We consider the quality of the manifold both at the vertex level and at the path level. To evaluate at different levels, we introduce two tasks: label retrieval and smooth path counting.

Label Retrieval. This task aims to see if nearby vertices belong to the same image class. It first selects a small subset of images as the database in a N-way-k-shot fashion. In this way, the selected images are evenly distributed in the manifold, and all images from under-represented classes, *i.e.*, classes with less than k samples, are excluded from the retrieval task. This removes the potential errors from under-training images in these classes, which is not the subject of this work. The rest images serve as the query images, and each image is used to find the most similar image from the database. A retrieve is then accurate *iff* the retrieved image has the same label as the query one. When retrieving with geodesic distances, some vertices can be unretrievable. We solve this problem by finding their nearest neighbors using the Euclidean distances. When more samples from the text domain are presented, they are used as privileged information, and thus their own label information is not computed.

Smooth Path Counting. In this task, we count the number of "smooth" paths in the graph G. We consider all shortest paths that start and end with an image vertex. This means that when text features, Φ, are used to construct the graph, they are only used to connect pairs of related images.

3.3. Retrieval in the Joint Space

To further utilize the textual information in a retrieval task, we consider the text-based image retrieval task. This task is similar to standard image retrieval, except that there are additional query inputs that describe the modifications from input images to target ones. To incorporate the queries, we need a compositional module that integrates information from both domains. This composition process can be seen as moving the image features along the direction learned from the queries. Such a task naturally favors the joint space, as it contains privileged text information from the captions. We thus propose to conduct the retrieval in the joint space, while keeping all other settings the same. In this way, this operation acts as a plug-and-play module that can be applied to various state-of-the-art text-based image retrieval works. We detailed this process next.

Compositional Module. Denote the input and target image features as ψ^i and ψ^t, their corresponding captions as ϕ^i and ϕ^t, and the query text features as ϕ^q, the goal for the compositional module is to learn a function f that combines ψ^i and ϕ^q such that the output resembles ψ^t most. We study the following two methods for f:

- **TIRG** [30] learns gating features and residual features from images and queries, and the output is the weighted sum of the two features. The residual features are learned by applying two 3x3 convolution layers with non-linearity on the concatenation of $[\psi^i, \phi^q]$. The gating function uses the same module followed by a Sigmoid function to learn a "filter" for the image features, *i.e.*, the output is the element-wise product of the filter signal and ψ^i.

- **Relationship** [26] is a VQA method that captures the relational reasoning. It forms a relational set by combining 2 local features $\psi^i(i, j)$ and $\psi^i(i', j')$ (at different locations) with text features ϕ^q. The set is passed through multi-layer perceptrons (MLPs), and the sum of the output is passed through another MLPs to get the final output.

Retrieval in the Joint Space. With the same assumption that there is a training mini-batch of B samples, $\{\psi^i_i, \psi^t_i, \phi^i_i, \phi^t_i, \phi^q_i\}_{i=1}^B$, we use the same cross entropy loss as (1):

$$L_{retrieval} = -\sum_{i=1}^{B} log\{\frac{exp\{\kappa(\psi^c_i, \psi^t_i)\}}{\sum_{j=1}^{B} exp\{\kappa(\psi^c_i, \psi^t_j)\}}\}, \quad (2)$$

where $\psi^c = f(\psi^i, \phi^q)$ is the composed features. We combine this loss with (1) to conduct text-based retrieval in the joint space:

$$L = L_{joint}(\psi^i, \phi^i) + L_{joint}(\psi^t, \phi^t) + \lambda * L_{retrieve}(\psi^c, \psi^t), \quad (3)$$

The first two terms ensure that the visual features are close to their corresponding textual features in the joint space, and the last term optimizes the compositional module. λ is set to 2 to balance the two training objectives. We empirically find that this setting is sufficient to have promising results. In our experiments, we simplify the text encoder to a standard LSTM in order to introduce minimum modifications over the original method. An identical text encoder is trained to encode the query texts separately.

4. Experimental Protocol and Results

To quantitatively evaluate the performance of the proposed manifold-based retrieval, we present the results on label retrieval tasks and text-based retrieval tasks. In both tasks, the evaluation metric is the recall at rank $K(R@K)$, as the percentage of test queries where the target image is within the top-K retrieval samples. For label retrieval tasks, we only consider $R@1$ because it resembles an image classification evaluation metric.

We use PyTorch in our experiments. The image encoder is the ResNet18 pretrained on Imagenet throughout all experiments. When constructing the graph, we use the latest USE (V4) in tensorflow-hub (embedding size 512), and the outputs are converted to PyTorch tensors. We adopt SGD optimizer with a starting learning rate of 0.01 with batch size of 50 for $200k$ iterations.

4.1. Datasets

Class Prediction. ADE20K [37] contains a wide range of objects in a variety of contexts. To ensure that each image has a precise label, we exclude images from the "Outlier" and "MISC" categories. This leads to 715 classes, and $17,956$ images. There are no corresponding captions for each image, but we can extract side information from the segmentation at the object level. OpenImage [21] is an image dataset annotated at different levels. For our purpose, we use the narratives for learning the joint space and image labels for testing. We use images from the validation set to train the model because the original training set ($\sim 9M$ images) is beyond our scope and all labels from the validation set are human-verified. In total, there are 528 classes, and $41,620$ images. Each image is annotated with multiple labels, and we consider the classification is correct *iff* all labels are matched. When doing class prediction, we select 3 images per class for both classes, resulting in $2,031$ and $1,486$ images with label information respectively.

Text-based Retrieval. We use Fashion-IQ [11], CSS [30], and a new synthetic dataset named CCI (see Section 4.2). These datasets contain attribute-like descriptions for images, and textual modification instructions between pairs of images. Fashion-IQ contains $18,000$ training samples. We use all the pairs that the side information for both images are available, which leads to $8,847$ samples. We evaluate the performance on the validation set.[1] CSS and CCI are two CLEVR-based datasets, where each image contains attribute information for all objects within the image, and the modification instructions are generated with templates. We follow the standard train-test splits [30] for CSS. Specifically, CSS has $18,012$ training samples and $18,057$ testing samples; CCI has $1,110$ training samples and $10,000$ testing samples.

4.2. CLEVR-Change-Iterative Dataset

Existing benchmarks focus on the scale and diversity of images. However, to provide a measurable definition of smoothness, it is more desirable to let every image be a variance of a "source" image, so that the difference between any two images can be categorical and thus countable. We use the CLEVR toolkit [17] to satisfy this goal. First, we generate one scene with random objects to serve as the "source" image. For this image we create ten "modified" images and the corresponding modification instructions following [23]. In particular, we apply *camera position changes* and *scene changes* besides the "distractor" operation to ensure that every image is unique. We iterate the modification process by considering the ten "modified" images as the "source" ones, and applying the same modification process on each of them, respectively. In total, we repeat the process four times, and generate $11,111$ images.

The formal definitions of "smooth" paths and valid (small and interpretable) differences are as follows:

- The difference between two images is valid if (a) two images only differ by a single attribute of a single object; or (b) two images only differ by one having a single additional object.
- We present the textual nodes as their corresponding images. Then we say a path is "smooth" if all adjacent vertices along this path are valid, and all non-adjacent vertices are not valid.

On average an image can form 9.73 valid pairs. We want non-adjacent vertices along a path to be invalid since otherwise we can ignore the vertices between two non-adjacent vertices to make the modification.

To use the dataset for the text-based retrieval task, we consider each modification providing a pair of source and target images and the corresponding modification instructions. We use the pairs from the first three iterations as the

[1] We evaluate on the val set since the ground truth for test set is not released.

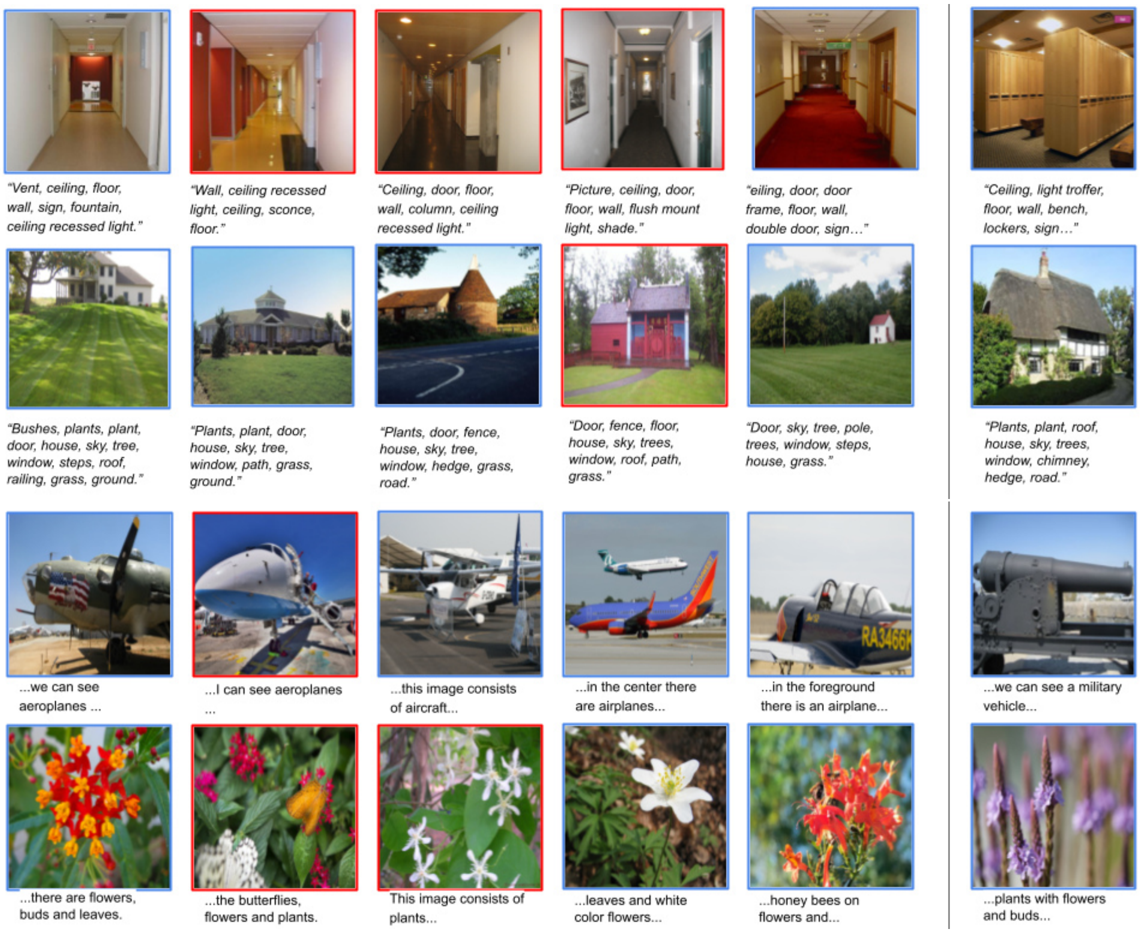
"Vent, ceiling, floor, wall, sign, fountain, ceiling recessed light."
"Wall, ceiling recessed light, ceiling, sconce, floor."
"Ceiling, door, floor, wall, column, ceiling recessed light."
"Picture, ceiling, door, floor, wall, flush mount light, shade."
"eiling, door, door frame, floor, wall, double door, sign…"
"Ceiling, light troffer, floor, wall, bench, lockers, sign…"
"Bushes, plants, plant, door, house, sky, tree, window, steps, roof, railing, grass, ground."
"Plants, plant, door, house, sky, tree, window, path, grass, ground."
"Plants, door, fence, house, sky, tree, window, hedge, grass, road."
"Door, fence, floor, house, sky, trees, window, roof, path, grass."
"Door, sky, tree, pole, trees, window, steps, house, grass."
"Plants, plant, roof, house, sky, trees, window, chimney, hedge, road."
...we can see aeroplanes ...
...I can see aeroplanes ...
...this image consists of aircraft...
...in the center there are airplanes...
...in the foreground there is an airplane...
...we can see a military vehicle...
...there are flowers, buds and leaves.
...the butterflies, flowers and plants.
This image consists of plants...
...leaves and white color flowers...
...honey bees on flowers and...
...plants with flowers and buds...

Figure 3: Qualitative examples from ADE20K (top 2 rows) and OpenImage (bottom 2 rows). We present the image and its corresponding text for each point. Image points are presented with blue boundaries, and text points' corresponding images are presented with red boundaries. Column 1 is the starting vertex, and column 5 is its nearest geodesic neighbor, with columns 2-4 presenting the shortest path. Column 6 is the nearest Euclidean neighbor.

training set, and the ones from the last iteration as the testing set. This results in $1,110$ training samples and $10,000$ testing samples.

4.3. Manifold Evaluation

As stated in Sec. 3.2.1, we evaluate the quality of a manifold at different levels with the label retrieval task and the smooth path counting task.

4.3.1 Label Retrieval

When retrieving the images in the target set, we can compute each points' geodesic neighbors or Euclidean neighbors, denoted as *+Geo* and *+Eu* respectively. When geodesic neighbors are computed, we report the recall scores both on retrievable points only, and on all points in which the unretrievable points are computed by Euclidean neighbors, denoted as *full*. We consider three feature collections to construct a graph. The first collection is the image features from the joint space that learning with the cross entropy loss (1), denoted as *Image*. This serves as a baseline for our experiment. To see the effectiveness of using texts to learn the visual-semantic joint space, we construct the second collection where image features are directly from the pretrained Resnet18, denoted as *Resnet*. Finally, we consider further utilizing texts by using both images and texts in the same joint space to construct the graph, and we denote this collection as *Joint*. For the baseline graph, we predict the labels by using either the Euclidean or the geodesic distances. For the other two graphs, we find the neighbors by geodesic distances only. Finally, we include a supervised model by adjusting the number of units for the final output layer of Resnet18 to match the number of classes. The model is trained on the database subset for a fair comparison. We use the standard cross-entropy loss for classification and the same settings (learning rate, optimizer, etc.) as

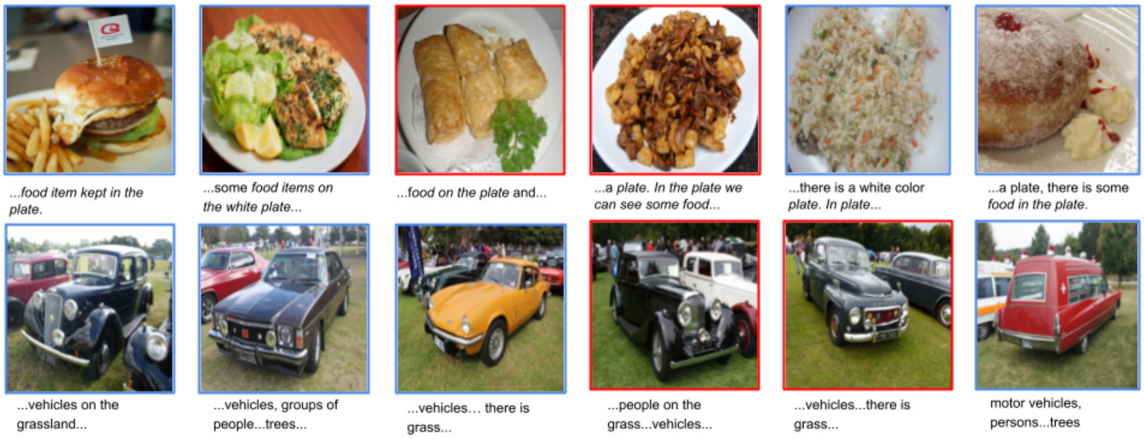
...food item kept in the plate.
...some food items on the white plate...
...food on the plate and...
...a plate. In the plate we can see some food...
...there is a white color plate. In plate...
...a plate, there is some food in the plate.
...vehicles on the grassland...
...vehicles, groups of people...trees...
...vehicles... there is grass...
...people on the grass...vehicles...
...vehicles...there is grass...
motor vehicles, persons...trees

Figure 4: Qualitative failure examples from OpenImage. Column 6 is now the nearest geodesic neighbor, with columns 2-5 presenting the shortest path. Row 1 shares a general concept, “foods on a plate,” and row 2 shares a general concept, “cars on grass.”

	Method	ADE20K		OpenImage	
		Accuracy	Retrievable Points	Accuracy	Retrievable Points
A	*Image + Eu*	0.1415	15925	0.0609	39665
B	*Resnet + Geo(full)*	0.1275	15925	0.0599	39665
	Resnet + Geo	0.1524	4169	0.1199	5569
C	*Image + Geo(full)*	0.1602	15925	0.0621	39665
	Image + Geo	0.2770	4169	0.1356	5569
D	***Joint + Geo (full)***	**0.1639**	15925	**0.0636**	39665
	Joint + Geo	0.2703	5774	0.1328	7461
E	*Supervised*	0.1718	15925	0.1302	39665

Table 1: Quantitative results of label prediction on ADE20K and OpenImage.

for training the joint space.

Table 1 summarizes the quantitative results on ADE20K and OpenImages. The supervised method (E) serves as the baseline when label information is available during the training. The comparison between using Euclidean distances and geodesic distances to find the neighbors (A vs. C) validates our statement that the geodesic distance is a better measurement for retrieval tasks. Moreover, the improvement by having additional texts to construct the graph (C vs. D) suggests the originally sparse visual feature space is now interpolated by the dense knowledge from the textual domain that is properly aligned with the visual domain. Note that the same mechanism works better on ADE20K; the reason can be that ADE20K provides side information at the object level, whereas OpenImage gives natural language captions that are likely to describe images at a higher level. Finally, we can see that the image features from the joint space give more robust representations than from the pretrained space (B vs. C). This motivates our experiment in Sec. 4.4.

4.3.2 Path Evaluation

We present some qualitative results of geodesic and Euclidean neighbors in Fig. 3, which also includes the shortest paths to the geodesic neighbors. We observe that traversing the graph with small steps allows for an image to find other images in the same neighborhood, while an Euclidean path may lead to an image from a different neighborhood. We include some qualitative results for unsuccessful prediction cases on ADE20K in Fig. 4 as well. The first example illustrates that nearby features from a visual-semantic joint space can sometimes be semantically similar but visually different. The second example reflects that the label retrieval task is only a loose representation for the robustness of feature spaces. Specifically, the starting image is labeled as “car, land vehicle” and the nearest geodesic neighbor is labeled as “land vehicle, limousine, van”. The inconsistency in labeling introduces additional errors.

In both successful and unsuccessful examples, we consistently observe that the changes between adjacent images is minor, even if they are from different domains. This in-

Method	Dress		Toptee		Shirt	
	$R@10$	$R@50$	$R@10$	$R@50$	$R@10$	$R@50$
TIRG [30]	0.1264	0.3386	0.1545	0.4080	0.1457	0.3690
TIRG + *in Joint*	**0.1507**	**0.3644**	**0.1856**	**0.4518**	**0.1638**	**0.4028**
Relation [26]	0.0744	0.2137	0.0918	0.2478	0.1084	0.2561
Relation + *in Joint*	**0.0843**	**0.2568**	**0.1147**	**0.2917**	**0.1178**	**0.2870**

Table 2: Retrieval Performance ($R@10$ and $R@50$) on Fashion IQ. The recall scores are higher when the method runs on the joint space. All results are from our implementation.

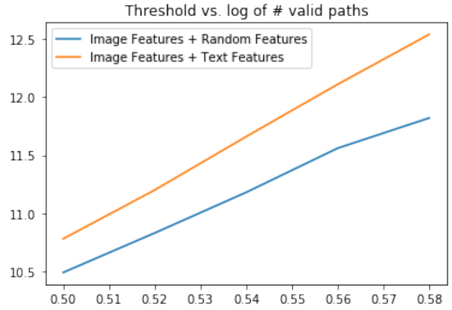

Figure 5: Log of number of paths under different thresholds and feature space. Note that under different thresholds adding text features leads to more smooth paths than adding random features do.

dicates that the additional vertices from the text domain are well-aligned with the original image vertices. To qualitative measure the wellness of this alignment, we perform the "smooth" path counting task on CCI. The result is shown in Fig. 5. As finding an optimal method to learn a well-aligned joint space is beyond the scope of our paper, we only compare adding text features from the same joint space with randomly generated features to show that having more semantically similar features results in more smooth paths.

4.4. Text-based Retrieval

As image features from a visual-semantic joint space are more robust, we can extend our work to text-based retrieval by adding a fully-connected layer after the encoders and one additional text encoder to encode side information. Our method is denoted as *in Joint* to emphasis that it only projects features into the cross-modal embedding without other modification to the original structure. Thus, our method can be used as a plug-and-play module. We also adopt the framework from [30] that allows different compositional methods. All networks are trained from scratch, except for the TIRG method on the CSS dataset, where a pretrained model is available. We only compare with the original work to show that our method works across different methods and on different datasets.

Table 2 shows $R@10$ and $R@50$ on the FashionIQ dataset under different categories, and Table 3 summaries $R@1$ performance on the CSS and CCI datasets. The performance of retrieval on the joint embedding space outperforms the one on the pretrained image space across different datasets. We note that the improvements are more significant on the CLEVR-based dataset than on FashionIQ. This is likely because the attributes on FashionIQ are less precise, *e.g.*, a multi-color shirt can refer to different combination of colors, while a red object is definite. Moreover, we note there is a large margin in the performance on CCI. As any two images in this dataset are similar (since all images are derived from the same original image), it is more challenging for an image encoder to learn a discriminative feature space, where images are better separated.

Method	CSS	CCI
TIRG [30]	0.7525	0.4123
TIRG *in Joint*	**0.7995**	**0.7802**
Relation [26]	0.5301	0.2121
Relation *in Joint*	**0.5531**	**0.6125**

Table 3: Retrieval performance ($R@1$) on CSS and CCI. The recall scores are higher when the method runs on the joint space. All results are from our implementation.

5. Conclusion and Future Work

We investigate the sub-optimal assumption that the relation between images can be approximated by negative Euclidean distance, and propose that a manifold structure and geodesic distances are more robust representations. We further study the manifold of a joint embedding space, where text points can be used as additional samples. These text samples are shown to benefit the image retrieval task, but we believe the application goes beyond this. For example, texts can connect visually different but semantically similar images, which can be useful to learn a relational network; they may be further incorporated into GCNs as extra neighborhood information. Overall, we show a way to represent the joint space of images and texts as a graph, opening multiple possibilities to interact with it.

References

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, 2018. 2

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual Question Answering. In *ICCV*, 2015. 2

[3] M. Belkin and P. Niyogi. Semi-supervised learning on riemannian manifolds. *Machine Learning*, 56:209–239, 2004. 1

[4] P. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14:239–256, 1992. 4

[5] Daniel Matthew Cer, Yinfei Yang, Sheng yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, C. Tar, Yun-Hsuan Sung, B. Strope, and R. Kurzweil. Universal sentence encoder. *ArXiv*, abs/1803.11175, 2018. 3

[6] O. Chapelle, J. Weston, and B. Schölkopf. Cluster kernels for semi-supervised learning. In *NIPS*, 2002. 2

[7] Yanbei Chen and Loris Bazzani. Learning joint visual semantic matching embeddings for language-guided retrieval. In *ECCV*, 2020. 3

[8] Fartash Faghri, David J. Fleet, J. Kiros, and S. Fidler. Vse++: Improving visual-semantic embeddings with hard negatives. In *BMVC*, 2018. 1, 3

[9] Zhe Gan, Chuang Gan, X. He, Y. Pu, K. Tran, Jianfeng Gao, L. Carin, and L. Deng. Semantic compositional networks for visual captioning. In *CVPR*, 2017. 2

[10] A. Gordo, J. Almazán, Jérôme Revaud, and Diane Larlus. Deep image retrieval: Learning global representations for image search. In *ECCV*, 2016. 3

[11] Xiaoxiao Guo, H. Wu, Yupeng Gao, Steven J. Rennie, and R. Feris. The fashion iq dataset: Retrieving images by combining side information and relative natural language feedback. *ArXiv*, abs/1905.12794, 2019. 5

[12] Xintong Han, Zuxuan Wu, Phoenix X. Huang, Xiao Zhang, Menglong Zhu, Y. Li, Yang Zhao, and L. Davis. Automatic spatially-aware fashion concept discovery. In *ICCV*, 2017. 1, 3

[13] David Harwath, A. Recasens, Dídac Surís, Galen Chuang, A. Torralba, and James R. Glass. Jointly discovering visual objects and spoken words from raw sensory input. In *ECCV*, 2018. 3

[14] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[15] Tobias Hinz, S. Heinrich, and Stefan Wermter. Semantic object accuracy for generative text-to-image synthesis. *IEEE transactions on Pattern Analysis and Machine Intelligence*, PP, 2020. 2

[16] X. Hu, Xi Yin, Kevin Lin, Longguang Wang, L. Zhang, Jianfeng Gao, and Zicheng Liu. Vivo: Surpassing human performance in novel object captioning with visual vocabulary pre-training. *ArXiv*, abs/2009.13682, 2020. 2

[17] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*, 2017. 2, 5

[18] V. Kazemi and Ali Elqursh. Show, ask, attend, and answer: A strong baseline for visual question answering. *ArXiv*, abs/1704.03162, 2017. 2

[19] Ira Kemelmacher-Shlizerman, E. Shechtman, R. Garg, and S. Seitz. Exploring photobios. In *SIGGRAPH*, 2011. 1

[20] Thomas Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017. 3

[21] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020. 2, 5

[22] Jiasen Lu, Jianwei Yang, Dhruv Batra, and D. Parikh. Neural baby talk. In *CVPR*, 2018. 2

[23] D. H. Park, T. Darrell, and A. Rohrbach. Robust change captioning. In *ICCV*, 2019. 5

[24] S. Reed, Zeynep Akata, Xinchen Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis. In *ICML*, 2016. 2

[25] Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and J. Hays. The sketchy database: learning to retrieve badly drawn bunnies. *ACM Trans. Graph.*, 35:119:1–119:12, 2016. 3

[26] Adam Santoro, D. Raposo, David G. T. Barrett, Mateusz Malinowski, Razvan Pascanu, P. Battaglia, and T. Lillicrap. A simple neural network module for relational reasoning. In *NIPS*, 2017. 4, 8

[27] Gunnar A. Sigurdsson, Jean-Baptiste Alayrac, Aida Nematzadeh, Lucas Smaira, Mateusz Malinowski, João Carreira, Phil Blunsom, and Andrew Zisserman. Visual grounding in video for unsupervised word translation. In *CVPR*, 2020. 3

[28] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP*, 2019. 2

[29] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319, 2000. 2

[30] Nam Vo, Lu Jiang, Chen Sun, Kevin Murphy, Li-Jia Li, Li Fei-Fei, and James Hays. Composing text and image for image retrieval-an empirical odyssey. In *CVPR*, 2019. 1, 3, 4, 5, 8

[31] J. Wang, Yang Song, Thomas Leung, C. Rosenberg, J. Philbin, Bo Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *CVPR*, 2014. 3

[32] L. Wang, Y. Li, J. Huang, and S. Lazebnik. Learning two-branch neural networks for image-text matching tasks. In *TPAMI*, 2018. 1, 3

[33] Liwei Wang, Yin Li, and Svetlana Lazebnik. Learning deep structure-preserving image-text embeddings. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5005–5013, 2016. 3

[34] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, R. Salakhutdinov, R. Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2017. 2

[35] Ting Yao, Yingwei Pan, Yehao Li, and T. Mei. Incorporating copying mechanism in image captioning for learning novel objects. In *CVPR*, 2017. 2

[36] Y. Zhang and H. Lu. Deep cross-modal projection learning for image-text matching. In *ECCV*, 2018. 3

[37] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017. 2, 5

[38] Dengyong Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf. Learning with local and global consistency. In *NIPS*, 2003. 2, 3