Expressional Region Retrieval

Xiaoqian Guo[1,2], Xiangyang Li[1,2], Shuqiang Jiang[1,2]
[1] Key Lab of Intelligent Information Processing, Institute of Computing Technology, CAS, Beijing, 100190, China
[2] University of Chinese Academy of Sciences, Beijing, 100049, China
{xiaoqian.guo,xiangyang.li}@vipl.ict.ac.cn,sqjiang@ict.ac.cn

ABSTRACT

Image retrieval is a long-standing topic in the multimedia community due to its various applications, e.g., product search and artworks retrieval in museum. The regions in images contain a wealth of information. Users may be interested in the objects presented in the image regions or the relationships between them. But previous retrieval methods are either limited to the single object of images, or tend to the entire visual scene. In this paper, we introduce a new task called expressional region retrieval, in which the query is formulated as a region of image with the associated description. The goal is to find images containing the similar content with the query and localize the regions within them. As far as we know, this task has not been explored yet. We propose a framework to address this issue. The region proposals are first generated based on region detectors and language features are extracted. Then the Gated Residual Network (GRN) takes language information as a gate to control the transformation of visual features. In this way, the combined visual and language representation is more specific and discriminative for expressional region retrieval. We evaluate our method on a new established benchmark which is constructed based on the Visual Genome dataset. Experimental results demonstrate that our model effectively utilizes both visual and language information, outperforming the baseline methods.

CCS CONCEPTS

• **Computing methodologies** → **Computer vision**; **Image representations**.

KEYWORDS

Expressional region retrieval, Gated residual network, Deep learning, Vision-Language

ACM Reference Format:

Xiaoqian Guo, Xiangyang Li and Shuqiang Jiang. 2020. Expressional Region Retrieval. In *28th ACM International Conference on Multimedia (MM'20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413567

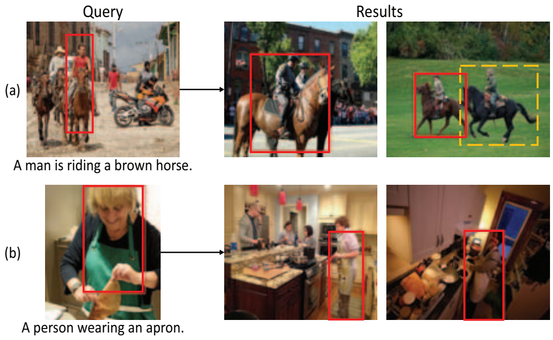


Figure 1: Two examples of the expressional region retrieval. The query is formulated as a region in an image with its language description. The results tend to be corresponding regions in image corpus.

1 INTRODUCTION

In recent years, as the amount of image data on the web has grown significantly, it becomes increasingly difficult to satisfy different user's requirements to find desired images. Users need to tell the system what kind of images they want, but how they describe their need as perfectly as possible still remains a key problem. They can use an image [10], a string [31], an object [8], a concept [2, 13], or combinations of the elements above to formulate their need as a query. The query is usually in a single modality, either visual or linguistic, which cannot depict the retrieval requirements satisfactorily. What's more, many works concentrate on global image retrieval which aims at finding images that are similar to the given query. Actually, the region is of much importance in image understanding and retrieval. Although previous works have tried to retrieve images based on local similarity, they have either focused only on local features [30] or on a single object in the image [12]. These methods do not take both visual and language information into account. In this work, we consider the case where the regions of images are concentrated on and the language information is also utilized.

In this work, we focus on the task of expressional region retrieval for images. An expressional region is an image region that can be expressed by natural language. It's better to use visual information and language information together, because they can complement each other. When it comes to visual content based image retrieval, only using visual information may lead to ambiguity. Sometimes, natural language can be used to provide more specific details. For example, in Figure 1 (a), the query region can be described with a sentence 'A man is riding a brown horse'. So the desired result would not include the yellow dotted region in the image on the

MM'20, October 12–16, 2020, Seattle, WA, USA.
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413567

Table 1: Comparison with different image retrieval tasks

Tasks	Query	Output	
		Results	Level
Text-Based Image Retrieval [38, 45]	A text string	Images that match the query	Image level
Scene Graph Based Image Retrieval [16]	A scene graph	Images related to the query semantically	Image level
Concept-Based Image Retrieval [2]	The concept layout	The objects that match the query layout	Object level
Instance Retrieval [34, 46]	An object	The same instance as the query	Object level
Class Retrieval [40]	An object	The object of the same class as the query	Object level
Near/Partial Duplicate Retrieval [19, 48, 50]	An object	The same object as the query	Object level
Expressional Region Retrieval	A region with a description	Regions that match the query	Region level

right, because the horse in it is black rather than brown. But if we delete the word 'brown' in the sentence, the yellow region becomes a positive result. Natural language can also help users to describe what they really want. As shown in Figure 1 (b), the user wants to retrieve regions that depict 'A person wearing an apron'. However, if the system only takes the query region as input, it's very likely to get results that satisfy 'A person in green'. That is not what the users really want. An image region that users are interested in and want to find similar ones may include multiple types of visual information, such as the attributes of objects, categories, and relationships between visual elements. The expression of the region not only describes the properties of the objects in the region, but also expresses the relationships between them. The visual information can eliminate the ambiguity of the language, and language can provide complementary information for visual scenes. Unlike a global image which contains a lot of useless background or misleading information, expressional region contains sufficient visual information and are less redundant. Compared to a single object in an image, an expressional region contains plentiful visual information.

We propose the task of expressional region retrieval: given an image and an expressional region (i.e., a region with a natural language description) as search query, we want to retrieve all images that contain the regions with the similar content as the query region and localize the regions in the images. The expressional region retrieval task attempts not only to retrieve related images, but also to localize the region within the image. The regions in expressional region retrieval task are more complex than those in traditional local retrieval task. As they are not limited to fixed categories of objects, but are more about multi-object interactions.

Expressional region retrieval task is different from other image retrieval tasks like text-based image retrieval [38, 45], instance retrieval [34], class retrieval [40] etc. Table 1 lists the comparison of our work with related works. Firstly, expressional region retrieval task is not a traditional cross-modal retrieval task like text-based image retrieval, which takes text that describes the image as query. The proposed retrieval task takes both visual and language information into consideration. Secondly, expressional region retrieval focuses on region level information not on image level or object level. Scene graph based image retrieval tends to find images that satisfy the query. Previous local image retrieval tasks only pay attention to limited object categories. For example, instance retrieval task [34, 46] aims at retrieving images containing the same object that may be captured under different views, illumination or with occlusions. And class retrieval task [40] is supposed to find images of the same class. In addition, the purpose of near/partial duplicate retrieval [19, 48, 50] is to make duplicate detection in dataset. Thirdly, our task focus on interactions between objects in the image region. Concept based image search [2, 21] takes the concept layout as query, where the concept can be formulated as tag, keyword, description or specific objects. The goal of concept based image search is to find objects that satisfy the query concept, and the spatial relationship of the objects needs to conform to the given layout. Although our task and concept based image retrieval both care about the interactions between multiple objects, our task is more flexible in the spatial relationship. We do not necessarily require that objects in the image region to be laid out exactly the same way as the query.

The challenges of expressional region retrieval lie in the following aspects. First, the expressional region retrieval task takes both visual and language information into consideration, which allows for more useful information to be utilized. So, how to integrate two different modalities in an effective and suitable way to make full use of the input information is worth considering. Second, we need to compute the similarity between the query regions and the candidate regions. How to make the feature representation more discriminative for better similarity measurement is also an issue. Third, the result we want for the retrieval task is the regions of images. The problem of how to find the target regions from the candidate images and localize them needs to be addressed.

In this paper, we propose a framework to solve the expressional region retrieval problem. First, we need to retrieve the target region from a set of candidate locations, which can come from proposal methods such as EdgeBox [51], Selective Search [41] or Region Proposal Network (RPN) [35]. In addition, we establish a new module called the Gated Residual Network (GRN) that integrates visual and language information to refine the visual representation for retrieval. Finally, we compute the similarity between the query region and all candidate regions, then sort them to obtain the results. Through experiments, we found that the feature matching methods for image retrieval lead to bad performance, as expressional region retrieval involves interactions and spatial relations between objects. Some existing methods involve jointly modeling visual and language information, such as simple concatenation and visual-semantic embedding [6]. But the complexity of computing the similarity between two different modalities results in poor performance. So we put forward a new idea that we need to utilize visual and language information at the same time. The language information and visual information have different roles, so they cannot be treated the same when combined. Simply merging or

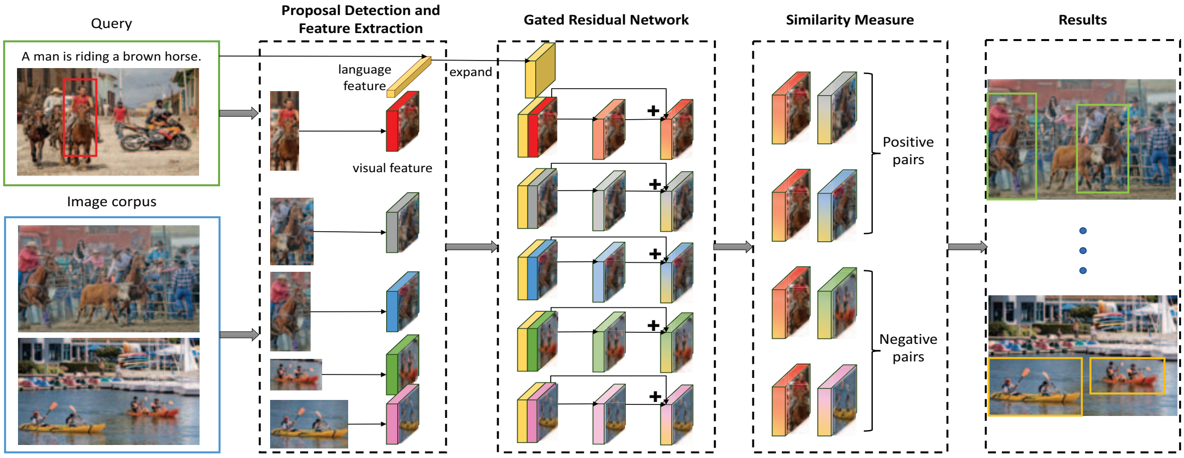


Figure 2: The framework of the expressional region retrieval. Given an image and a region in it with a description as query, we first detect region proposals from image corpus and extract visual and language features. Then the Gated Residual Network processes the representation by integrating visual and language features. The similarity measure module takes these proposals as input and outputs the similarities of the query and candidate proposals. Finally, we sort the candidate proposals by the similarity score.

mapping two kind of information into the common space fails to take into account the different roles they play. And what we want is to refine the visual representation guided by the language information, as we believe the language is complementary to the visual information. The key idea behind the GRN model is that the expression of the region is used as a gate to control what information of the image region should be propagated, so that we can easily find regions with similar expressions and visual characteristics. We achieve this goal by using a gated residual connection to make the control mechanism work.

We establish a new benchmark for expressional region retrieval task. It is based on the Visual Genome(VG) dataset [17]. The experimental results illustrate that our proposed method outperforms exiting methods in this benchmark. To summarize, we make the following contributions:

(i) We introduce a new retrieval task called expressional region retrieval, which takes an image and a region of it with the associated description as query. It aims at retrieving images containing the similar content as the query and localizing the regions within them.

(ii) We establish a new benchmark based on the Visual Genome dataset for the expressional region retrieval task. The benchmark can be used for related experiments of expressional region retrieval, as well as for image region matching task.

(iii) We propose a framework for expressional region retrieval task. The Gated Residual Network (GRN) model is introduced for integrating visual and language information to make the visual representation more discriminative for measuring similarity. Experimental results validate the effectiveness of the proposed method.

2 RELATED WORK

Image Retrieval. Image retrieval is an important multimedia problem, it aims to retrieve relevant images from the image corpus given a query. At first, images were retrieved by the visual cues, such as texture and color [23]. Extracting global descriptors [7] is a straightforward method. But global descriptors may be inefficient when images are changed. Thus, the local feature based image retrieval has attracted much attention. Many previous works focus on instance-level image retrieval, such as instance/object image retrieval [14, 34, 40, 46], near/partial duplicate image retrieval [19, 48, 50] and logo retrieval [36]. SIFT-based methods [24] and deep learning based methods [18, 29] are used to address the problem. Different from previous image retrieval tasks, the proposed expressional region retrieval task focuses on the region level not on image level or object level.

Cross-Modal Image Retrieval. Cross-modal retrieval means that the query and the content to be retrieved are not in the same modality. Cross-modal image retrieval takes various types of query to retrieve images. For example, text to image retrieval [38, 45, 47] is the retrieval of images given sentence query. Tag/concept based image retrieval [2, 21] uses the tag of objects or the concept to find corresponding images. In addition, sketch to image retrieval [5], keyword to image retrieval [39], scene graph retrieval [16] and cross view image retrieval [22] have attracted attention recently. The mainly difference between cross-modal image retrieval and our expressional region retrieval is that we do not refer to a different modality from visual information as the query. The query in the proposed task contains both visual and language information.

Referring Expressions. Referring expressions have attracted research interest in multimedia and related areas. There are two tasks which are closely related to referring expressions (i.e., comprehension and generation). The task of referring expression comprehension requires a system to localize the object described by a given expression. Generative models [12, 26], embedding models [33, 45] and attention mechanism [3] are used to locate the target object region when given the query language. The referring

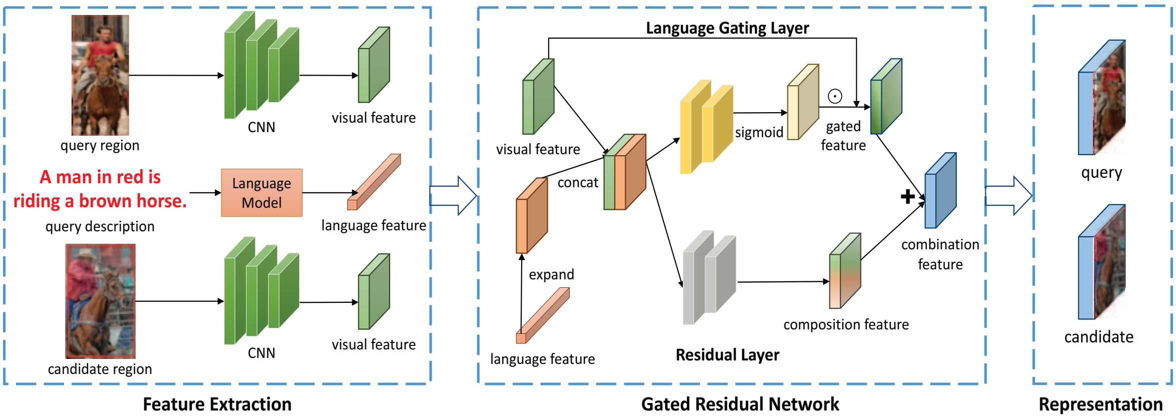


Figure 3: Overview of the proposed Gated Residual Network. The components of the GRN are language gating layer and residual layer. The GRN model takes the visual feature and language feature as input and outputs a combination feature that integrates the two input features.

expression generation task is to generate a natural language expression for a specified object within an image. Many approaches have been proposed for generation task [11, 25]. In order to generate more accurate descriptions, unlike most of the works only using the whole images as the context, there are some methods adding the instance-level visual information to obtain the context features [20, 49]. The proposed expressional region retrieval task is different from the referring expression comprehension task mainly in the following two aspects. First, expressional region retrieval takes both the visual region within an image and the corresponding language as input, while referring expression comprehension only utilizes the query language as input. The proposed task is more challenge as it needs to integrate information from different modalities to represent the queries comprehensively. Second, referring expression comprehension is trying to ground the query into a region of an image, while the expressional region retrieval task aims to find the corresponding regions from a set of images.

3 METHOD

Our proposed framework for expressional region retrieval is illustrated in Figure 2. First, the proposal detection and feature extraction module outputs candidate proposals and corresponding features. Then the GRN module takes these features as input to integrate visual information together with the language. This module makes the visual features enhanced by the language features. Finally, given the refined representation, the similarity measure module computes the similarities between the query region and candidate regions. We will discuss each module in this section.

3.1 Proposal Detection and Feature Extraction

Candidate region proposals are firstly generated by the proposal detection module. There are many popular proposal generation methods and we utilize the supervised RPN [35] in our framework to get region proposals.

The feature extraction module is used to extract features from the input including images and language. The module contains two streams, one for visual information and another for language information. As shown in Figure 3 (left), for visual input, we use a convolutional neural network to obtain the spatial feature vector for the query region and the candidate proposals generated from images in the corpus. We adopt ResNet-18 without the last layer to get the visual feature vector $X \in \mathbb{R}^{\mathbb{W}\times\mathbb{H}\times\mathbb{C}}$, where W is the width, H is the height, and $C = 512$ is the number of feature channels.

For language input, we encode the expression of the query region using a language model to get the feature vector $L \in \mathbb{R}^{\mathbb{D}}$. We compare two different language models for this task. The first one is a standard LSTM architecture. Word representations are individually passed through a LSTM cell, each producing their own hidden state. LSTM can model the relationship between words in a sentence and maintain the word ordering. In this way, we define the feature vector L to be the hidden state at the final time step whose size D is 512. The second language model is a Self-Attention Language model [1]. The model uses the vector obtained by the weighted sum of each word embedding to represent the sentence. As shown in Figure 4, the model takes into account the contextual information in the language, while we take the mean pooling of each word as the context. The context vector is concatenated with all word embeddings. It is then passed through a fully connected layer with Softmax. So we can get the context score of each word. Then the inner product of these weights and the original word embeddings is used to produce the final representation of the given sentence, which can be regarded as a context weighted sum vector.

3.2 Gated Residual Network

The GRN module is illustrated in Figure 3, which takes the visual features and language features together to produce the final representation. In this module, we aim to integrate the visual features with the language features. Different with traditional multi-modal image retrieval methods, we do not want to map the two kind of information into the same space or simply merge them with each other. We prefer to keep the visual features in its original embedding space. So we try to use the language features to control the

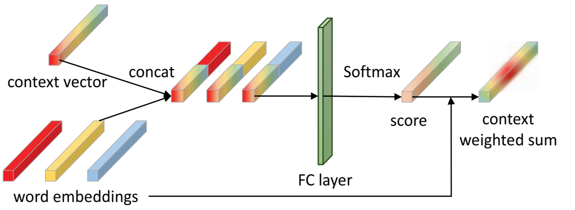


Figure 4: Self-Attention Language Model. A model which builds a weighted context sum representation for the given sentence.

enhancement of the visual features. Inspired by [9, 27], we use a gated residual connection to achieve this goal.

1) Language Gating Layer. The language gating layer transforms the input visual feature representation X and the input language features L into a new representation Y_g with the formulation as

$$Y_g = \sigma(W_{g2} * Relu(W_{g1} * [X, L])) \odot X \tag{1}$$

where σ is the element-wise sigmoid function, $\odot$ is element-wise product, $[X, L]$ means concatenating the feature X and the feature L, $*$ represents 2d convolution and W_{g1}, W_{g2} are 3x3 convolution filters which are trainable parameters. $\sigma(W_{g2} * Relu(W_{g1} * [X, L])) \in [0, 1]$ represents a set of learned gates which are applied to the input visual features.

Our goal is to refine the visual features with the help of language. We consider the gate mechanism to allow the network to control what visual information should be enhanced according to the language. Compared to previous gate mechanism, our language gating mechanism takes the features of two modalities of visual and language as input. We apply language gating layer on the query region and candidate region with the same expression, and then the visual features that match the expression have been strengthened. So that we can make the visual representation more discriminative. The language gating layer can capture dependencies among visual elements together with the language. For example, the query describes 'a man is skiing', and the candidate region shows a skiing person, snow and some trees. In this example, trees might be less important where the snow and skiing person is crucial for the retrieval task. The language gating layer can learn to reduce the weight of unimportant part and increase the weight of strongly related visual parts. We utilize this architecture to learn discriminative features.

2) Residual Layer. Residual connections have been proved to be useful in some visual tasks [9]. They demonstrate faster and better training process as well as better performance to some extent. We apply a residual connection which can be formulated as:

$$y = F(x, \{W_i\}) + f(x) \tag{2}$$

where x and y are the input and output features. The function $F(x, \{W_i\})$ represents the mapping to be learned and $f(x)$ means a transformation for the input feature x.

According to the above description, the function $F(x, \{W_i\})$ in Equation 2 is regarded as the language gating layer, where the input vector x represents the visual feature X and the language feature L. So the Equation 2 can also be written as:

$$y = F(X, L, \{W_i\}) + f(X, L) \tag{3}$$

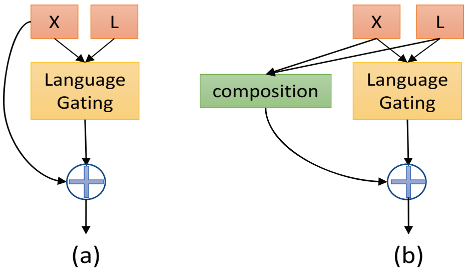


Figure 5: Several designs for the transformation function in residual layer. X denotes the input visual feature and L the input language feature. The two shortcuts are (a) Visual Transformation (b) Compose Transformation. The Language Gating module is described in Section 3.2.

Several designed architectures are shown in Figure 5 for the transformation function $f(X, L)$. The **Visual Transformation** (Figure 5(a)) directly uses the original image feature X as the residual feature. The Equation 3 can be written as $y = F(X, L, \{W_i\}) + X$. This shortcut only takes the language information into account in the language gating procedure. The **Compose Transformation** (Figure 5(b)) combines the visual features and language features together. The composition result can be computed by:

$$Y_r = f(X, L) = W_{r2} * Relu(W_{r1} * ([X, L])) \tag{4}$$

where W_{r1}, W_{r2} represent the convolution operation.

The GRN module finally takes the gated feature and the composition feature together as:

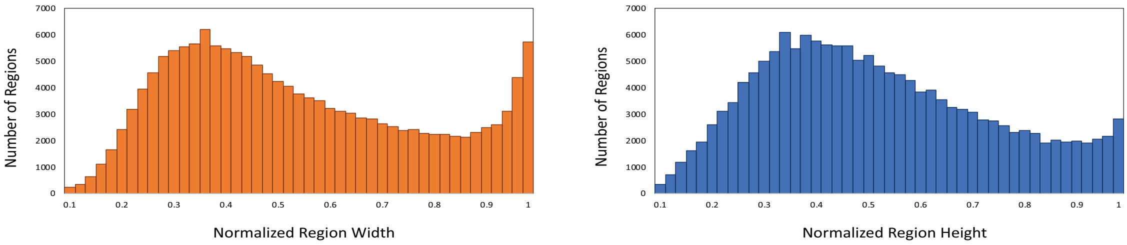
$$Y_{final} = w_g Y_g + w_r Y_r \tag{5}$$

Different from the traditional residual connection, we apply two weights to adjust the effect of the gated feature and the composition feature for the final representation. The w_g, w_r in Equation 5 are learnable weights to balance two types of features. And we set $w_g = 1, w_r = 1$ at beginning. The intuition of this module is that we want to enhance the visual representation to better distinguish the positive pairs from the negative ones, instead of simply fusing two-modal features. The gated mechanism makes it possible to keep the input visual feature and output feature in the same meaningful feature space [43]. And then the residual connection is added to finetune the features in this feature space.

The Equation 1 and Equation 4 show the operation which is applied to the convolutional layer, the visual feature $X \in \mathbb{R}^{\mathbb{W} \times \mathbb{H} \times \mathbb{C}}$ denotes the feature map from CNN. So we expand the language feature $L \in \mathbb{R}^{\mathbb{D}}$ along the width and height dimension to make its shape compatible to the visual feature. Meanwhile, we can alternatively apply the operation on the fully connected layer, where W=H=1 for visual feature X. In this way, the $W_{g1}, W_{g2}, W_{r1}, W_{r2}$ change from convolution to linear projection.

3.3 Similarity Measure

The similarity measure module takes the refined feature to learn the similarities between the query region and candidate region. We employ distance between two features to measure their similarity. In this module, the network computes matching loss of valid pairs



(a) (b)

Figure 6: Distributions of the width and height of the region normalized by the image size.

to learn discriminative features. The training objective is to push closer the query region and positive candidate regions, while pulling apart the features of non-similar regions.

We have a training minibatch of B queries, ϕ_i is the final representation of the query region, and φ_i^+ is the representation of the positive region for the query. Meanwhile, there are N-1 negative examples $\varphi_1^-, \ldots, \varphi_{N-1}^-$ of the query. The loss function is formulated as

$$L = \frac{-1}{B}\sum_{i=1}^{B} log(\frac{Sim(\phi_i, \varphi_i^+)}{\sum_{j=1}^{N} Sim(\phi_i, \varphi_j^-)}) \tag{6}$$

where Sim is the similarity function which is implemented as the cosine distance in our experiments. When the value of N in Equation 6 is set to 2, the loss function is equal to the soft triplet loss [44].

3.4 Training and Inference

We minimize the loss function as Equation 6 in the training procedure. During training, the query text simultaneously controls the visual features of query region and candidate regions. The negative examples of a given query are sampled from the positive regions for other queries in the same minibatch.

While in the inference and retrieval procedure, the query text not only affects the query, but also refines the visual representation of all candidate regions generated from the images. After the representation has been enhanced, we compute the similarity between the query region and each candidate region as the score, and then sort all candidate regions by their scores.

4 EXPERIMENTS

We evaluate our proposed method for expressional region retrieval and demonstrate its necessity and superiority to several previous methods. We perform our study on a new benchmark we established based on the large scale Visual Genome dataset (VG) [17].

4.1 Dataset

Since existing benchmarks for image retrieval cannot be used for expressional region retrieval task, we create a new one based on the Visual Genome dataset. The VG dataset consists of about 108077 images, with every image including an average of 50 regions described by a phrase or sentence [17]. The descriptions tend to be highly diverse and can focus on a single object or multiple objects. They encompass the most salient parts of the image while also capturing the background.

For the proposed expressional region retrieval task, we expect the image regions to contain richer information, so we first remove the image region only describing a single object or with no object, like 'A bag' or 'The blue sky'. According to [17], the regions that cover large portions of the image tend to be general descriptions of an image while regions that cover only a small fraction of the image tend to be more specific. What we want is image regions with abundant information. So we have to avoid regions with too little information that are too small, and also do not need too large regions close to the whole image. Therefore we delete the regions with the area less than one-tenth of the whole image and larger than nine-tenth. In this way, the regions left tend to contain multiple objects and at least one kind of relationship between objects, such as 'A man taking a photo of the elephant' or 'Tall buildings surrounded by trees'. Furthermore, regions of very similar sizes and locations in the same image often contain highly similar information. So we compute the Intersection over Union (IoU) ratio between two regions in one image. If the IoU is larger than 0.9, we only keep the one with the larger area. In Figure 6, we show the distribution of the region width/height normalized by the entire image. We see that the majority of the regions tend to be around 30% to 50% of the entire image. We observe an increase in the number of regions when the width of region is close to the entire image, but not the height. The reason may be that there are often regions that span the entire image in the horizontal direction, but rarely in the vertical direction.

Next, we begin to look for similar regions and label them as a pair. We determine their similarity based on the matching of the region descriptions. To ensure the diversity for the image regions, we use both BLEU score [32] and METEOR score [4]. We first compute the BLEU score between two descriptions:

$$S_{bleu} = N\sum_{n=1}^{N} p_n(d_1, d_2) \tag{7}$$

where d_1, d_2 refer to two region descriptions and p_n calculates the percentage of n-grams matched in two descriptions. We set N=2 in Equation 7 to ensure the diversity of similar regions. Then we compute the METEOR score between two descriptions with the BLEU score higher than 0.7.

$$S_{meteor} = M(d_1, d_2) \tag{8}$$

where $M(d_1, d_2)$ represents the METEOR score between d_1 and d_2. We keep the positive pair when the METEOR score is higher than the threshold 0.6.

Table 2: The performance of expressional region retrieval. 'GT' represents that the region proposals are got from ground truth bounding boxes and 'RPN' represents that the region proposals are generated with the pretrained RPN.

	R@1		R@10		R@50		R@100		med r	
	GT	RPN	GT	RPN	GT	RPN	GT	RPN	GT	RPN
Visual Feature	0.341	0.171	2.817	1.756	9.606	6.015	15.946	10.726	19	28
Visual-Language LSTM Feature	0.422	0.183	3.068	1.744	8.2	5.848	12.637	7.992	27	32
Visual-Language Concat Feature	0.534	0.323	3.063	1.915	11.358	7.186	20.442	15.772	18	20
GRN w/o Residual Layer (Ours)	1.184	0.688	4.328	2.211	14.447	10.824	24.394	19.39	13	16
GRN (Ours)	**1.21**	**0.707**	**4.66**	**2.507**	**16.913**	**12.965**	**27.741**	**22.086**	**10**	**15**

In total, we generate 35K images with 116K regions for train, 5K images with 20K regions for validation and 5K images with 21K ground truth regions for test. The training set contains 64K query regions, while the validation set and the test set contain 5012 and 4967 query regions respectively. Each image we used contains at least one positive region. Each query consists of a query region with a language description, and several target regions. To be specific, we integrate similar regions into a set according to the pairs generated above. Then we randomly select one region from a set as the query region where the rest is the target. We apply this process to training set, validation set and test set respectively.

4.2 Experimental results

Experimental Setup. The main metric we used for retrieval is recall at rank k (R@K), which is computed as the percentage of test queries where at least one target is within the top K retrieved results. We evaluate top 1, 10, 50, 100 results in our experiments. What's more, the median rank of the target image region (med r) is also calculated, the lower the better. We complete our experiments in PyTorch. We use ResNet-18 pretrained on ImageNet [37] as the image encoder, and the Word2Vec [28] embedding pretrained on Google News as the word embedding. We use the words that appear more than once as the vocabulary and replace other words with an <UNK> tag. The batch size B in Equation 6 is set to 32.

We conduct our experiments on two kind of proposal settings. The first one is that the region proposals are got from ground truth bounding boxes. When in the training procedure, we use 3 negative examples for each query, where the negative ones are randomly selected from the regions in the same minibatch but are not positive for the given query. The second proposal setting is that the region proposals are generated by detectors. We follow DensecCap [15] and pretrain a separate RPN on the training data. The pretrained RPN is followed by a RNN language model so that the regions detected by it contain more information than proposals generated by object detectors. We use the proposals got from the pretrained RPN for training and inference. In this setting, the evaluation is simply performed by measuring the IoU ratio between a ground truth box and the extracted one. In training procedure, only if the IoU is larger than 0.7, the region is considered to be true positive. And we use the false positive regions as negative examples. In the inference procedure, we regard the region as positive when the IoU is larger than 0.5.

The experiments are carried out on a workstation with E5-2620, 64GB RAM, and GeForce GTX TITAN X. It takes 600ms to get retrieval results for one query, where the time cost is obtained by averaging over the whole test set.

Baseline Model. We compare our methods with some popular approaches for multi-modal image retrieval. For a fair comparison, we train all methods including ours using the same pipeline, with the only difference being in the GRN module.

(a) **Visual Feature.** Features of the last fully connected layer are extracted for each proposal. Then we directly use the extracted visual features without the query text to calculate the similarity scores. In this method, language information is not used.

(b) **Visual-Language LSTM Feature.** We use a CNN and a standard LSTM model to combine visual and language features. The visual features for each proposal are extracted from a CNN. Then the LSTM is used to encode the image and text by inputting the visual features in the first time step, following by words in the query text [42]. We take the final state of the LSTM as the final combined representation.

(c) **Visual-Language Concat Feature.** In this setting, we first extract the visual features for each proposal and encode the query text. Then we concatenate the visual features and language features. After passing it through a fully connected layer, we use the features to measure similarity

The results are shown in Table 2 and some qualitative results are shown in Figure 7. For fair comparisons, same proposals are used for each method. It can be observed that our method which uses the gate mechanism works much better than the baselines. We see that most methods that take language information into account outperform the methods that only consider visual information. But there are exceptions that the effect of the Visual-Language LSTM Feature may be even worse than the Visual Feature. Though we input the visual features into the Visual-Language LSTM Feature at first, the characteristic of the LSTM may result in the language information occupying a more important position. Our method gets a gain when the region proposals are ground truth (GT) and also generated with the region detectors (RPN). Our approaches using the same proposals get better performance than the baselines. The results indicate that the language information plays a role in controlling the transformation of visual information rather than mapping to the same embedding space.

4.3 Ablation Studies

In this section, we report the results of various ablation studies, to explore which parts of our approach matter the most.

Language models. As shown in Table 3, the language model used to embed query text information may influence the retrieval result. The performance of LSTM is not as good as the Self-Attention model. When we retrieve for image regions in the dataset which are similar to the query region and query text, the order of words

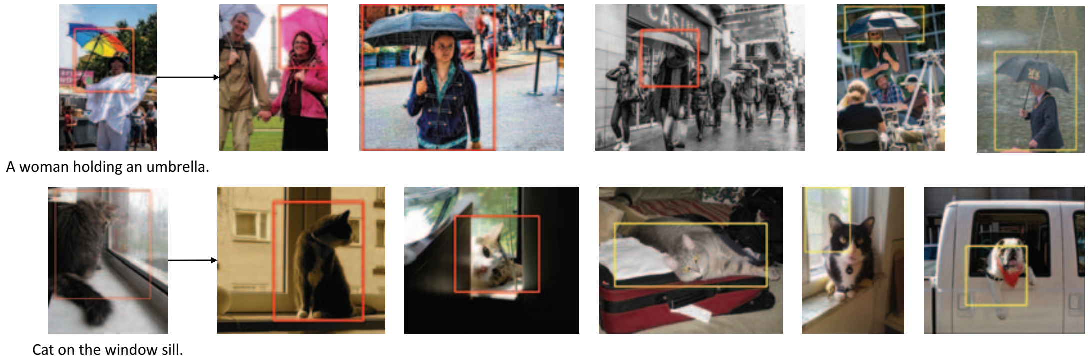


Figure 7: Some retrieval examples. The red region refers to the positive results and the yellow ones are negative.

Table 3: The retrieval performance of different language model.

model	R@1	R@10	R@50	R@100	med r
LSTM	0.526	4.236	14.887	26.604	**10**
Self-Attention	**1.21**	**4.66**	**16.913**	**27.741**	**10**

Table 4: The retrieval performance of different residual transformation.

model	R@1	R@10	R@50	R@100	med r
w/o residual layer	1.184	4.328	14.447	24.394	13
Visual	1.004	3.893	12.817	21.273	15
Compose	**1.21**	**4.66**	**16.913**	**27.741**	**10**

Table 5: The retrieval performance of different dimension of the feature.

model	R@1	R@10	R@50	R@100	med r
fc	1.07	4.472	15.494	25.919	11
conv	**1.21**	**4.66**	**16.913**	**27.741**	**10**

in the description may not matter much. But the important words, like the word that refers to the salient object, an action or spatial relationship may influence the results more. The LSTM may be more concerned about the sequential characteristic about the description, but ignore the importance of the salient objects. What's more, LSTM is too complicated to converge, but the Self-Attention model is more efficient and performs better.

Residual transformation. According to Table 4, the best result is from the Compose Transformation of residual layer, indicating that the residual connection is beneficial. What's more, it can be observed that combining the visual and language information together in a proper way performs better than the original visual feature as the composition feature. Actually, directly using the visual feature is worse than without a residual layer. This could because the original visual feature is not enhanced by the language information, so the usage of it will cause a reduction in the result.

Conv or FC. We can apply the GRN module on last fc layer or last convolutional layer (feature map). Table 5 compares the effect of these two approaches. When the operation is applied to the last convolution layer, it improves the performance. We believe that's because the feature map contains more information than fc layer, such as spatial information, which may be useful for expressional region retrieval.

5 CONCLUSIONS

In this paper, we introduce a new image retrieval task, expressional region retrieval, where the query is formulated as an image region with a natural language description. The proposed retrieval task focuses on region-level information and tends to find regions with similar content as the query from image corpus. Based on the observation that visual and language information should be both taken into account to benefit retrieval, we introduce a Gated Residual Network (GRN) model to integrate two kinds of information. Our approach can properly deal with the combined visual and language representation, which is more comprehensive, specific and discriminative for expressional region retrieval. Furthermore, we establish a new benchmark based on the Visual Genome dataset for expressional region retrieval task. And our method achieves promising results compared to baseline methods.

Our work for expressional region retrieval is relatively preliminary. First, the interrelationship between image and region needs to be considered. We will take contextual information into account to better understand regional content in future work. Thus, we use proposals generated in advance for retrieval. If the proposal detection process is carried out guided by the query, the retrieval results may be more satisfying. In this way, the speed of retrieval is becoming an issue. Exploiting the hashing methods will improve the speed and boost the performance of retrieval.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China under Grant 61532018 and U1936203, in part by the National Postdoctoral Program for Innovative Talents under Grant BX20200338.

REFERENCES

[1] A. Burns, R. Tan, K. Saenko, S. Sclaroff, and B. Plummer. 2019. Language Features Matter: Effective Language Representations for Vision-Language Tasks. In *IEEE*

International Conference on Computer Vision. 7473–7482.

[2] T. Chua, H. Pung, G. Lu, and H. Jong. 1994. A Concept-Based Image Retrieval System. In *Hawaii International Conference on System Sciences*, Vol. 3. 590–598.

[3] C. Deng, Q. Wu, Q. Wu, F. Hu, F. Lyu, and M. Tan. 2018. Visual Grounding via Accumulated Attention. In *IEEE Conference on Computer Vision and Pattern Recognition.* 7746–7755.

[4] M. Denkowski and A. Lavie. 2014. Meteor Universal: Language Specific Translation Evaluation for Any Target Language. In *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation.* 376–380.

[5] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa. 2011. Sketch-Based Image Retrieval: Benchmark and Bag-of-Features Descriptors. *IEEE Trans. Vis. Comput. Graph.* 17, 11 (2011), 1624–1636.

[6] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov. 2013. DeViSE: A Deep Visual-Semantic Embedding Model. In *Proceedings of the 26th International Conference on Neural Information Processing Systems.* 2121–2129.

[7] A. Gordo, J. Almazán, J. Revaud, and D. Larlus. 2016. Deep Image Retrieval: Learning Global Representations for Image Search. In *European Conference on Computer Vision.* 241–257.

[8] S. Guadarrama, E. Rodner, K. Saenko, N. Zhang, R. Farrell, J. Donahue, and T. Darrell. 2014. Open-vocabulary Object Retrieval. In *Robotics Science and Systems.*

[9] K. He, X. Zhang, S. Ren, and J. Sun. 2016. Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition.* 770–778.

[10] S. C. H. Hoi, W. Liu, M. R. Lyu, and W. Y. Ma. 2006. Learning Distance Metrics with Contextual Constraints for Image Retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 2. 2072–2078.

[11] R. Hu, M. Rohrbach, J. Andreas, T. Darrell, and K. Saenko. 2017. Modeling Relationships in Referential Expressions with Compositional Modular Networks. In *IEEE Conference on Computer Vision and Pattern Recognition.* 4418–4427.

[12] R. Hu, H. Xu, M. Rohrbach, J. Feng, K. Saenko, and T. Darrell. 2016. Natural Language Object Retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition.* 4555–4564.

[13] X. Hua and G. Qi. 2008. Online Multi-Label Active Annotation: Towards Large-Scale Content-Based Video Search. In *Proceedings of the 16th ACM International Conference on Multimedia.* 141–150.

[14] S. Jiang, S. Liang, C. Chen, Y. Zhu, and X. Li. 2019. Class Agnostic Image Common Object Detection. *IEEE Trans. Image Process.* 28, 6 (2019), 2836–2846.

[15] J. Johnson, A. Karpathy, and L. Fei-Fei. 2016. DenseCap: Fully Convolutional Localization Networks for Dense Captioning. In *IEEE Conference on Computer Vision and Pattern Recognition.* 4565–4574.

[16] J. Johnson, R. Krishna, M. Stark, L. Li, D. A. Shamma, M. S. Bernstein, and L. Fei-Fei. 2015. Image Retrieval using Scene Graphs. In *IEEE Conference on Computer Vision and Pattern Recognition.* 3668–3678.

[17] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L. Li, D. A. Shamma, M. S. Bernstein, and L. Fei-Fei. 2017. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *Int. J. Comput. Vis.* 123, 1 (2017), 32–73.

[18] K. Li, G. Qi, J. Ye, T. Yusuph, and K. A. Hua. 2016. Supervised Ranking Hash for Semantic Similarity Search. In *IEEE International Symposium on Multimedia.* 551–558.

[19] L. Li, S. Jiang, Z. Zha, Z. Wu, and Q. Huang. 2013. Partial-Duplicate Image Retrieval via Saliency-Guided Visual Matching. *IEEE Multim.* 20, 3 (2013), 13–23.

[20] X. Li and S. Jiang. 2018. Bundled Object Context for Referring Expressions. *IEEE Trans. Multimedia* 20, 10 (2018), 2749–2760.

[21] X. Li, T. Uricchio, L. Ballan, M. Bertini, C. G. Snoek, and A. D. Bimbo. 2016. Socializing the Semantic Gap: A Comparative Survey on Image Tag Assignment, Refinement, and Retrieval. *ACM Comput. Surv.* 49, 1, Article 14 (2016), 39 pages.

[22] T. Lin, Y. Cui, S. Belongie, and J. Hays. 2015. Learning Deep Representations for Ground-to-Aerial Geolocalization. In *IEEE Conference on Computer Vision and Pattern Recognition.* 5007–5015.

[23] G. Liu and J. Yang. 2013. Content-Based Image Retrieval Using Color Difference Histogram. *Pattern Recognit.* 46, 1 (2013), 188–198.

[24] D. G. Lowe. 2004. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* 60, 2 (2004), 91–110.

[25] R. Luo and G. Shakhnarovich. 2017. Comprehension-Guided Referring Expressions. In *IEEE Conference on Computer Vision and Pattern Recognition.* 3125–3134.

[26] J. Mao, J. Huang, A. Toshev, O. Camburu, A. Yuille, and K. Murphy. 2016. Generation and Comprehension of Unambiguous Object Descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition.* 11–20.

[27] A. Miech, I. Laptev, and J. Sivic. 2017. Learnable pooling with Context Gating for video classification. *CoRR* abs/1706.06905 (2017).

[28] T. Mikolov, W. Yih, and G. Zweig. 2013. Linguistic Regularities in Continuous Space Word Representations. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics.* 746–751.

[29] J. Y. Ng, F. Yang, and L. S. Davis. 2015. Exploiting Local Features from Deep Networks for Image Retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops.* 53–61.

[30] H. Noh, A. Araujo, J. Sim, T. Weyand, and B. Han. 2017. Large-Scale Image Retrieval with Attentive Deep Local Features. In *IEEE International Conference on Computer Vision.* 3476–3485.

[31] V. Ordonez, X. Han, P. Kuznetsova, G. Kulkarni, M. Mitchell, K. Yamaguchi, K. Stratos, A. Goyal, J. Dodge, A. Mensch, et al. 2016. Large Scale Retrieval and Generation of Image Descriptions. *Int. J. Comput. Vis.* 119, 1 (2016), 46–59.

[32] K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics.* 311–318.

[33] B. A. Plummer, A. Mallya, C. M. Cervantes, J. Hockenmaier, and S. Lazebnik. 2017. Phrase Localization and Visual Relationship Detection with Comprehensive Image-Language Cues. In *IEEE International Conference on Computer Vision.* 1946–1955.

[34] A. S. Razavian, J. Sullivan, S. Carlsson, and A. Maki. 2016. Visual Instance Retrieval with Deep Convolutional Networks. *ITE Trans. MTA* 4, 3 (2016), 251–258.

[35] S. Ren, K. He, R. Girshick, and J. Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems.* 91–99.

[36] J. Revaud, M. Douze, and C. Schmid. 2012. Correlation-Based Burstiness for Logo Retrieval. In *Proceedings of the 20th ACM International Conference on Multimedia.* ACM, 965–968.

[37] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* 115, 3 (2015), 211–252.

[38] S. N. Chowdhury, M. Malinowski, A. Bulling, and M. Fritz. 2016. Xplore-M-Ego: Contextual Media Retrieval Using Natural Language Queries. In *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval.* 243–247.

[39] B. Siddiquie, R. S. Feris, and L. S. Davis. 2011. Image Ranking and Retrieval based on Multi-Attribute Queries. In *IEEE Conference on Computer Vision and Pattern Recognition.* 801–808.

[40] L. Torresani, M. Szummer, and A. Fitzgibbon. 2010. Efficient Object Category Recognition Using Classemes. In *European Conference on Computer Vision.* 776–789.

[41] J. R. Uijlings, K. E. Sande, T. Gevers, and A. W. Smeulders. 2013. Selective Search for Object Recognition. *Int. J. Comput. Vis.* 104, 2 (2013), 154–171.

[42] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. 2015. Show and Tell: A Neural Image Caption Generator. In *IEEE Conference on Computer Vision and Pattern Recognition.* 3156–3164.

[43] N. Vo, L. Jiang, C. Sun, K. Murphy, L. Li, L. Fei-Fei, and J. Hays. 2019. Composing Text and Image for Image Retrieval - an Empirical Odyssey. In *IEEE Conference on Computer Vision and Pattern Recognition.* 6432–6441.

[44] N. N. Vo and J. Hays. 2016. Localizing and Orienting Street Views Using Overhead Imagery. In *European Conference on Computer Vision.* 494–509.

[45] L. Wang, Y. Li, and S. Lazebnik. 2016. Learning Deep Structure-Preserving Image-Text Embeddings. In *IEEE Conference on Computer Vision and Pattern Recognition.* 5005–5013.

[46] S. Wang and S. Jiang. 2015. INSTRE: A New Benchmark for Instance-Level Object Retrieval and Recognition. *ACM Trans. Multim. Comput. Commun. Appl.* 11, 3 (2015).

[47] Z. Wang, X. Liu, H. Li, L. Sheng, J. Yan, X. Wang, and J. Shao. 2019. CAMP: Cross-Modal Adaptive Message Passing for Text-Image Retrieval. In *IEEE International Conference on Computer Vision.* 5763–5772.

[48] Z. Wu, Q. Ke, M. Isard, and J. Sun. 2009. Bundling Features for Large Scale Partial-Duplicate Web Image Search. In *IEEE Conference on Computer Vision and Pattern Recognition.* 25–32.

[49] L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg. 2016. Modeling Context in Referring Expressions. In *European Conference on Computer Vision.* 69–85.

[50] W. Zhou, Y. Lu, H. Li, Y. Song, and Q. Tian. 2010. Spatial Coding for Large Scale Partial-Duplicate Web Image Search. In *Proceedings of the 18th ACM International Conference on Multimedia.* 511–520.

[51] C. L. Zitnick and P. Dollar. 2014. Edge Boxes : Locating Object Proposals from Edges. In *European Conference on Computer Vision.* 391–405.