RetrieveGAN: Image Synthesis via Differentiable Patch Retrieval

Hung-Yu Tseng*[1], Hsin-Ying Lee*[1], Lu Jiang[2], Weilong Yang[2], Ming-Hsuan Yang[1,2]

[1]University of California, Merced [2]Google Research

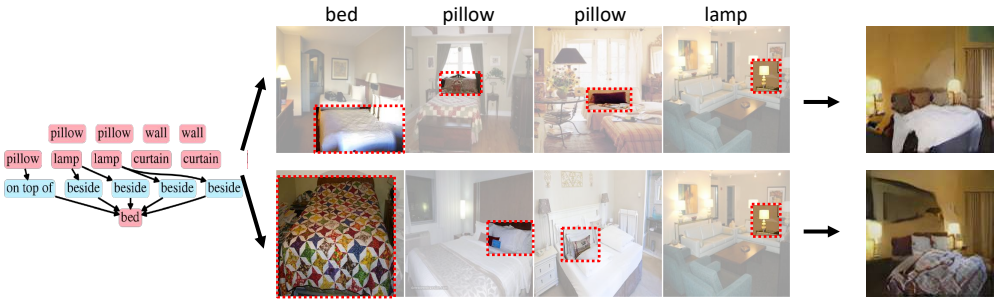


Figure 1. **Image synthesize from retrieved examples.** We propose the RetrieveGAN model that takes as input the scene graph description and learns to 1) select mutually compatible image patches via a differentiable retrieval process and 2) synthesize the output image from the retrieved patches.

Abstract

Image generation from scene description is an essential task for controlled generation, which is beneficial to content creation and image editing. In this work, we aim to synthesize images from scene description with retrieved patches as reference. We propose a differentiable retrieval module. With the differentiable retrieval module, we can (1) Make the entire pipeline end-to-end trainable, enabling the learning of better feature embedding for retrieval. (2) Encourage the selection of mutual compatible patches with additional objective functions. We conduct extensive quantitative and qualitative experiments to demonstrate that the proposed method can generate realistic and diverse images, where the retrieved patches are mutually compatible.

1. Introduction

Image generation from scene descriptions has received considerable attention. Taking advantage of generative adversarial networks (GANs), recent research employs conditional GAN for the image generation task. There are various conditional contexts such as scene graph [4], bounding box [11], and text [8]. A stream of work has been driven by parametric models that rely on the network to capture and model the appearance of objects [4]. The other stream explores the semi-parametric model that leverages a memory bank to aim the synthesizing process [9].

In this work, we focus on the semi-parametric models that a memory bank is provided for the retrieval purpose. Existing retrieval-based image synthesis methods have two issues. First, the retrieval process usually requires pre-defined embeddings. Since the retrieval process is non-differentiable, the pre-defined embeddings are isolated from the generation process and thus cannot guarantee the retrieved objects are suitable given the large variations of different datasets. Second, there are usually multiple objects to be retrieved given a description. However, the conventional retrieval process selects each patch independently and cannot take the mutual relationship into consideration.

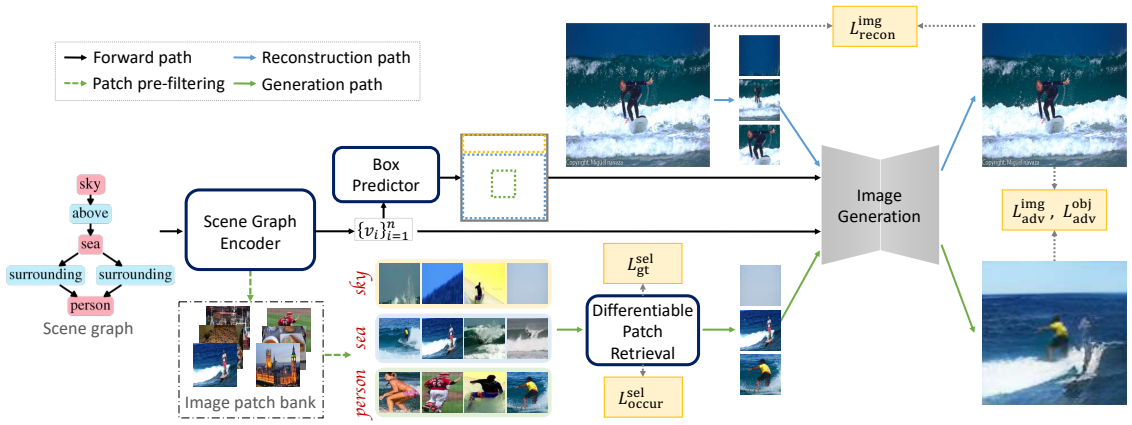
We propose *RetrieveGAN*, an image generation framework with a differentiable retrieval process. With the proposed differentiable retrieval design, the proposed RetrieveGAN is capable of retrieving image patches that 1) considers the surrogate image generation quality, and 2) are mutually compatible for synthesizing a single image.

We evaluate the proposed methods through extensive qualitative, quantitative experiments, and user preference study. With the proposed approach, we show that 1) the generated images are realistic, and 2) the retrieved patches are mutually compatible.

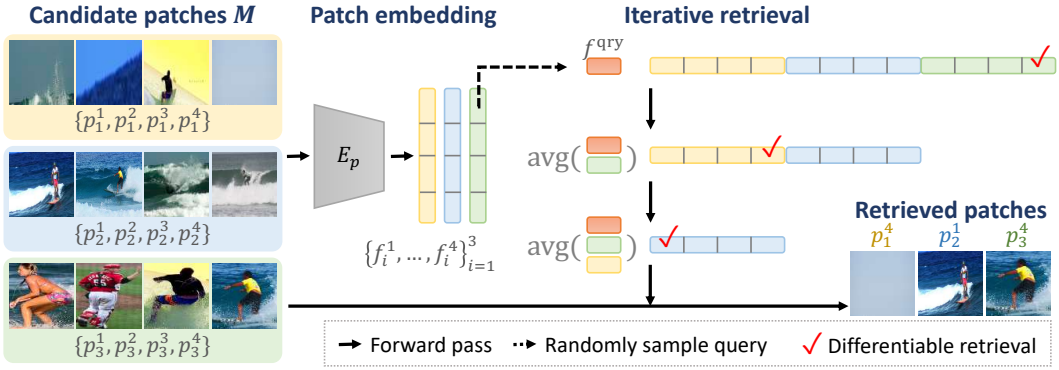
2. RetrieveGAN

Our goal is to synthesize a realistic image $x \in \mathbb{R}^{H \times W \times 3}$ from the input scene graph g by compositing appropriate image patches retrieved from the image patch bank. As

*Equal contribution.



(a) RetrieveGAN



(b) Iterative differentiable patch retrieval

Figure 2. **Method overview.** (a) The approach takes as input the scene graph description and sequentially performs scene graph encoding, patch retrieval, and image generation to synthesize the desired scene image. (b) Given a set of candidate patches, we first extract the corresponding patch features using the patch embedding function. We then randomly select a patch feature as the query feature for the iterative retrieval process. At each step of the iterative procedure, we select the most compatible patch compared to the already selected patches. The iteration ends as all the objects are assigned with a selected patch.

the overview shown in Figure 2, the proposed RetrieveGAN framework consists of three stages: scene graph encoding, patch retrieval, and image generation. We adopt the strategies in the PasteGAN [9] approach for the scene graph encoding and image generation stages, while introduce the proposed patch retrieval phase as follows.

2.1. Patch Retrieval

The patch retrieval aims to select a number of mutually compatible patches for synthesizing the final image. We illustrate the overall process on the bottom side of Figure 2. We first pre-filtered the candidate patches $\{M(o_i)\}_{i=1}^{n}$ for each object o_i using the pre-trained graph convolutional network features in sg2im [4]. We then use a patch embedding function E_p to extract the patch features. Starting with a randomly sampled patch feature as a query, we propose an iterative retrieval process to select compatible patches for all objects. In the following, we first describe how a single differentiable retrieval is operated. The proposed iterative retrieval process is then introduced. Finally, we illustrate the objective function used to facilitate the training of the patch retrieval module.

Differentiable retrieval for a single object. Given the query feature f^{qry}, we aim to sample a single patch from the candidate set $M(o) = \{p^1, p^2, \cdots, p^k\}$ for object o. Let $\pi \in \mathbb{R}^k_{>0}$ be the categorical variable with probabilities $P(x = i) \propto \pi_i$ which indicates the probability of selecting the i-th patch from the bank. To compute π_i, we calculate the ℓ_2 distance between the query feature and the corresponding patch feature, namely $\pi_i \propto -\|f_{\text{qry}} - E_p(p^i; \theta_{E_p})\|_2$, where E_p is the embedding function and θ_{E_p} is the learnable mode parameter. The intuition is that the candidate patch with smaller feature distance to the query feature should be sampled with higher probability. Through learning θ_{E_p}, we hope our model to retrieve compatible patches guided by our loss functions. As we are sampling from a categorical distribution, we use the Gumbel-Max trick [3] to sample a single patch:

$$\arg\max_i [P(x = i)] = \arg\max_i [g_i + \log \pi_i] = \arg\max_i [\hat{\pi}_i], \quad (1)$$

where $g_i = -\log(-\log(u_i))$ is the re-parameterization term and $u_i \sim \text{Uniform}(0,1)$. To make above process differentiable, the argmax operation is approximated with the continuous softmax operation: $s = \text{softmax}(\hat{\pi}) = \frac{\exp(\hat{\pi}_i/\tau)}{\sum_{q=1}^{k}\exp(\hat{\pi}_q/\tau)}$, where τ is the temperature controlling the degree of the approximation.

Iterative differentiable retrieval for multiple objects. Rather than retrieving only a single image patch, the proposed framework needs to select a subset of n patches for the n objects defined in the input scene graph. We therefore adopt the weighted reservoir sampling method [7] to perform the subset sampling from the candidate patch sets. Without loss of generality, denote $M = \{p_i | i = 1, \ldots, n \times k\}$ as the multiset (with possible duplicated elements) consisting of all candidates patches in which n is the number of objects, and k is the size of each candidate patch set.

We first compute the vector $\hat{\pi}_i$ defined in (1) for all patches. We then iteratively apply n softmax operations over $\hat{\pi}$ to approximate the top-k selection. Let $\hat{\pi}_i^{(j)}$ denote the probability of sampling patch p_i at iteration j and $\hat{\pi}_i^{(1)} \leftarrow \hat{\pi}_i$. The probability is iteratively updated by:

$$\hat{\pi}_i^{(j+1)} \leftarrow \hat{\pi}_i^{(j)} + \log(1 - s_i^{(j)}), \tag{2}$$

where $s_i^{(j)} = \text{softmax}(\hat{\pi}^{(j)})_i$. Essentially, (2) sets the entry of selected patch to negative infinity thus ensuring this index will not be selected again. After n iterations, we compute the relaxed n-hot vector $s = \sum_{j=1}^{n} s^{(j)}$, where $s_i \in [0,1]$ indicates the score of selecting the i-th patch, and we have $\sum_{i=1}^{|M|} s_i = n$. The entire process is differentiable with respect to the model parameters (*i.e.* θ_{E_p}).

We make several modifications to the iterative process based on practical consideration. First, our candidate multiset $M = \{p_i\}_{i=1}^{n \times k}$ is formed by n groups of pre-filtered patches for n objects. In addition, to incorporate the prior knowledge that compatible images patches tend to lie closer in feature space, we adopt a greedy strategy to encourage selecting image patches that are compatible with the already selected ones. We detail this process in Figure 2(b).

2.2. Training Objective Functions

Ground-truth selection loss. As the ground-truth patches are available at the training stage, we add them to the candidate set M. Given one of the ground-truth patch features as the query feature f^{qry}, the ground-truth selection loss $L_{\text{gt}}^{\text{sel}}$ encourages the retrieval process to select the other ground-truth patches from the same image.

Co-occurrence penalty. We design a co-occurrence loss to ensure the mutually compatible between the retrieved patches. Given a set of retrieved patches, we minimize the pair-wise distance on the co-occurrence space, namely

$$L_{\text{occur}}^{\text{sel}} = \sum_{i,j} d(F_{\text{occur}}(p_i), F_{\text{occur}}(p_j)), \tag{3}$$

where the mapping function F_{occur} is pre-trained via contrasting learning.

Table 1. **Quantitative comparisons.** The first row shows the results of models that predict bounding boxes during inference time. The second row shows the results of models that take ground-truth bounding as inputs during inference time.

Datasets	COCO-Stuff			Visual Genome		
	FID ↓	IS ↑	DS ↑	FID ↓	IS ↑	DS ↑
sg2im [4]	136.8	4.1±0.1	0.02±0.0	126.9	5.1±0.1	0.11±0.1
AttnGAN [8]	72.8	8.4±0.2	0.14±0.1	114.6	**10.4**±0.2	0.27±0.2
PasteGAN [9]	59.8	8.8±0.3	**0.43**±0.1	81.8	6.7±0.2	**0.30**±0.1
RetrieveGAN	**43.2**	**10.6**±0.6	0.34±0.1	**70.3**	7.7±0.1	0.24±0.1
sg2im (GT)	79.9	8.5±0.1	0.02±0.0	111.9	5.8±0.1	0.13±0.1
layout2im [11]	45.3	10.2±0.6	0.29±0.1	**44.0**	9.3±0.4	**0.29**±0.1
PasteGAN (GT)	54.9	9.6±0.2	**0.38**±0.1	68.1	6.7±0.1	0.28±0.1
RetrieveGAN (GT)	**42.7**	**10.7**±0.1	0.21±0.1	46.3	**9.1**±0.1	0.23±0.1
real data	6.8	24.3±0.3	-	6.9	24.1±0.4	-

Domain adversarial loss. We use two discriminators D_{img} and D_{obj} to encourage the realism of the generated images on the image-level and object-level, respectively.

$$\begin{aligned} L_{\text{adv}}^{\text{img}} &= \mathbb{E}_x[\log D_{\text{img}}(x)] + \mathbb{E}_{\hat{x}}[\log(1 - D_{\text{img}}(\hat{x}))], \\ L_{\text{adv}}^{\text{obj}} &= \mathbb{E}_p[\log D_{\text{obj}}(p)] + \mathbb{E}_{\hat{p}}[\log(1 - D_{\text{obj}}(\hat{p}))], \end{aligned} \tag{4}$$

where x and p are respectively denoted as the real image and patch, while $\hat{x}$ and $\hat{p}$ respectively represent the generated image and the patch crop from the generated image.

Bounding box regression loss L_{bbx}. We penalize the prediction of the bounding box coordinates with the $\ell 1$ distance.

Image reconstruction loss $L_{\text{recon}}^{\text{img}}$. Given the ground-truth patches and the ground-truth bounding box coordinates, we use the $\ell 1$ distance to encourage the generation module to reconstruct the ground-truth image.

The full loss functions for training our model is:

$$\begin{aligned} L = &\lambda_{\text{gt}}^{\text{sel}} L_{\text{gt}}^{\text{sel}} + \lambda_{\text{occur}}^{\text{sel}} L_{\text{occur}}^{\text{sel}} + \lambda_{\text{adv}}^{\text{img}} L_{\text{adv}}^{\text{img}} + \lambda_{\text{recon}}^{\text{img}} L_{\text{recon}}^{\text{img}} + \\ &\lambda_{\text{adv}}^{\text{obj}} L_{\text{adv}}^{\text{obj}} + \lambda_{\text{bbx}} L_{\text{bbx}}, \end{aligned} \tag{5}$$

where λ controls the importance of each loss term.

3. Experimental Results

Datasets. We use the standard scene generation benchmark datasets, COCO-Stuff [1] and Visual Genome [5], in all experiments. Except for the image resolution which is 128×128, we follow the protocol in sg2im [4] to preprocess and split the dataset.

Evaluated methods. We compare the proposed approach

Figure 3. **Retrieved patches.** For each sample, we show the retrieved patches which are used to guide the following image generation process. We also show the original image of each selected patch for more clear visualization.

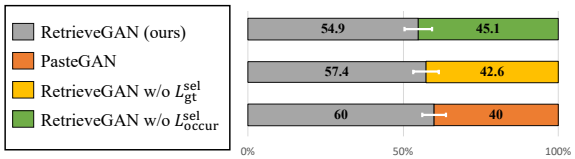


Figure 4. **User study.** We conduct user the study to evaluate the mutual compatibility of the selected patches.

to the sg2im[4], AttnGAN [8], layout2im [11], and PasteGAN [9] schemes in the experiments.

Evaluation Metrics. We use the IS [6] (realism), FID [2] (realism and diversity), DS [10] (Diversity) scores as the evaluation metrics.

3.1. Quantitative Evaluation

Realism and diversity. We conduct the evaluation using two different settings. First, bounding boxes of objects are predicted by models. Second, ground-truth bounding boxes are given as inputs in addition to the text or scene graph. The results of these two settings are shown in the first and second row of Table 1, respectively. Since the patch retrieval process is optimized to consider the generation quality during the training stage, our approach performs favorably against the other algorithms in terms of realism.

Patch compatibility. To evaluate the compatibility between the retrieved patches, we conduct a user study. For each scene graph, we present two sets of patches selected by different methods, and ask user "which set of patches are more mutually compatible and more likely to exist in the same image?". Figure 4 presents the results of the user study. The proposed method outperforms PasteGAN, which uses the pre-defined patch embedding function to perform the retrieval. The results also validate the usefulness of the ground-truth selection loss and the co-occurrence loss.

3.2. Qualitative Evaluation

To better visualize the source of retrieved patches, we present the generated images as well as the original images of selected patches in Figure 3. The proposed method can tackle complex scenes where multiple objects are present. With the help of selected patches, each object in the generated images has a clear and reasonable appearance (*e.g.* the boat in the second row and the food in the third road). Most importantly, the retrieved patches are mutually compatible thanks to the iterative retrieval process with the differentiable retrieval module. As shown in the first example in Figure 3, the selected patches are all related to baseball. In contrast, the PasteGAN method has chances to select irrelevant patches (*i.e.* the soccer player).

4. Conclusions

In this work, we propose a differentiable retrieval module to aid the image synthesis from the scene description. The differentiable property enables the module to learns a better embedding function with the image generation process. Moreover, through the iterative process, the retrieval module selects mutually compatible patches as reference for the generation. Qualitative and quantitative evaluations validate that the synthesized images are realistic while the retrieved patches are compatible.

References

[1] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *CVPR*, 2018. 3

[2] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, 2017. 4

[3] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. 2017. 2

[4] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *CVPR*, 2018. 1, 2, 3, 4

[5] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 123(1):32–73, 2017. 3

[6] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *NIPS*, 2016. 4

[7] Sang Michael Xie and Stefano Ermon. Reparameterizable subset sampling via continuous relaxations. 2019. 3

[8] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *CVPR*, 2018. 1, 3, 4

[9] LI Yikang, Tao Ma, Yeqi Bai, Nan Duan, Sining Wei, and Xiaogang Wang. Pastegan: A semi-parametric method to generate image from scene graph. In *NeurIPS*, 2019. 1, 2, 3, 4

[10] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 4

[11] Bo Zhao, Lili Meng, Weidong Yin, and Leonid Sigal. Image generation from layout. In *CVPR*, 2019. 1, 3, 4