Controllable and Progressive Image Extrapolation

Anonymous WACV submission

Paper ID 685

Abstract

Image extrapolation aims at expanding the narrow field of view of a given image patch. Existing models mainly deal with natural scene images of homogeneous regions and have no control of the content generation process. In this work, we study conditional image extrapolation to synthesize new images guided by the input structured text. The text is represented as a graph to specify the objects and their spatial relation to the unknown regions of the image. Inspired by drawing techniques, we propose a progressive generative model of three stages, i.e., generating a coarse bounding-boxes layout, refining it to a finer segmentation layout, and mapping the layout to a realistic output. Such a multi-stage design is shown to facilitate the training process and generate more controllable results. We validate the effectiveness of the proposed method on the face and human clothing dataset in terms of visual results, quantitative evaluations, and flexible controls.

1. Introduction

Given an image patch with a narrow field of view, image extrapolation aims at expanding it by generating plausible visual content outside the image boundaries. The extrapolation is a challenging task since it requires to synthesize new content that aligns well with the given image patch. To the best of our knowledge, only a few approaches [29, 52, 36] have been developed to address this topic, and all are designed for *unconditional* extrapolation where the target image is generated solely based on the input patch. This is often achieved by finding low-level cues of similar patterns from the given image or external databases. These methods perform well on natural images of homogeneous regions.

A core problem, however, is that oftentimes a user has
some concept in mind from which one wants to generate an
image, and the most straightforward way to express the concept is via text. Consider an example in Figure 1(a), for the
given patch, users may have different ideas of extrapolating
the lower body, wearing the dress or pants. An ideal model
should directly take both the patch and text into account to



Figure 1. Definition and motivation of the extrapolation task. (a) Conditional image extrapolation takes the input of the image patch and text. Users may want to synthesize the lower body to generate the *dresses* or *pants* object and can control the generation by the text input. (b) Top: illustration of human layout drawing in the coarse-to-fine manner. Bottom: intermediate and final outputs of our progressive generation model, which corresponds to each step of human layout drawing.

generate the target image.

In this paper, we study conditional image extrapolation where the inputs are an image patch and a structured text that specifies desired properties to synthesize. The image patch serves as the same role as that in the unconditional extrapolation, whereas the input text controls the content generation outside the image boundaries. Similar to [16], we represent the structured text as a scene graph to circumvent handling the ambiguity in natural languages. The scene graph [23, 41, 40, 16] consists of nodes to represent objects and edges to describe their relations (spatial arrangements in our case). Conditional image extrapolation offers more flexibility than existing counterparts in that users can control what and where to generate outside the image boundaries, thereby allowing users to generate a variety of target images from the same image patch with different text descriptions. Our problem is related to text-to-image generation [51, 50, 44] but differs in its usage of multimodal input of both image and text.

A straightforward solution to this problem is to learn a deep generative model (*e.g.*, [13, 37, 25, 6]) to directly translate unknown regions to plausible RGB pixels. However, this approach is likely to generate blurry images of poor quality. More importantly the text cannot effectively

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146 147

148

149

150

151

152

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

control the generated content. The reason is that learning such a direct mapping between two different modalities (from text to high-dimensional pixel space) is extremely difficult. As a result, the current key research question for conditional image extrapolation is how to make the image generation process controllable by the input text and amicable to the input image patch.

To address this issue, we mimic the process of how an 116 painter creates an artwork. Before filling out the details, 117 a painter often progressively refine a sketch from object 118 contours to finer layouts, as shown on the top row of Fig-119 ure 1(b). Motivated by this, we propose a progressive gen-120 erative model that consists of three stages to extrapolate an 121 image patch. We first generate a *bounding-box layout* from 122 the scene graph to roughly indicate the size and spatial loca-123 tion of each object. Conditioned on the bounding-box lay-124 out, we then learn to generate a semantic segmentation lay-125 out, where each pixel is represented as an object class label. 126 Finally, we map the segmentation layout to the extrapolated 127 pixels via image-to-image translation. See the bottom row 128 of Figure 1(b). These modules are first separately trained 129 for individual tasks and then jointly optimized. 130

We evaluate the conditional image extrapolation on two public datasets in terms of visual results, quantitative evaluations and flexible controls. Extensive experimental results demonstrate that our model performs favorably against existing methods. The progressive training not only speeds up the convergence substantially but also makes the generated content more controllable. In addition, the intermediate outputs, byproducts of our model, are semantically meaningful to users. The main contributions of this work are summarized as follows:

- We study a new task of conditional image extrapolation which takes multimodal inputs of image and text.
- We propose an effective progressive generative network to synthesize new content outside image boundaries by generating layouts as sub-tasks.
- We realize controllable extrapolation to generate diverse extrapolated images which respect different indications in the scene graph.

2. Related Work

153 Image extrapolation. Early extrapolation algorithms generally follow a retrieve-and-compose strategy where an ex-154 155 ternal library of sample images that depict the similar scene 156 is assumed to be available. For example, Efros and Freeman [8] expand the small texture patch with similar patches 157 and develop an optimal boundary with minimum cost for 158 composition. By extending similar textured patches to im-159 160 ages of the similar scene category, Zhang et al. [52] extrap-161 olate photos by utilizing the self-similarity of a reference image to generate a set of local transformations. To handle different viewpoints and appearance variations, a few methods [29, 36] use library images to search good candidates and align them with the given input. However, those nonparametric methods are mainly limited in semantically new content and requiring proper reference databases. With the recent advances of generative models [9, 24], a few neural network based methods greatly improve the performance. For example in texture synthesis, Zhou *et al.* [54] directly train a feed-forward network to expand a certain small texture patch to a larger one. Other methods [39, 46, 34] focus on single object or scene images. However, they are still under the uncontrollable setting.

Image inpainting. Compared to extrapolation, image inpainting concerns filling the unknown regions inside the image. A number of image inpainting methods [20, 48, 12, 22, 38] learn to fill the holes inside the image with different design of architectures and losses, and achieve better results over diffusion-based [4, 33] or patch-based [2] schemes. However, those approaches seldom pay attention to extrapolation explicitly where the number of unknown pixels is much more than that of known pixels.

Text-to-image generation. Our problem is also related to text-to-image generation which aims at synthesizing image content only from text descriptions. Much progress has been made in this field in improve the quality of results in higher resolution [51, 50, 53, 11], reduce the ambiguity in text with attention mechanism [42, 19] or other text representations such as the scene graph [16, 47]. Conditional image extrapolation is apparently different from text-to-image generation since the former input contains both image patch and text. Therefore, conditional image extrapolation poses an unique challenge that is how to align the generated image with the input patch controlled by the text which is not concerned in text-to-image generation approaches.

Curriculum and progressive learning. Our progres-199 sive training approach is related to curriculum learning 200 schemes [3], which aim to master a complex job by first 201 learning easier aspect of the task and gradually take more 202 complex samples into consideration. It has been widely 203 used to weight training samples [14, 5] or to prioritize the 204 tasks in multi-task learning [30, 26]. The line of research 205 work generally regards finding an optimal order of execut-206 ing some known tasks [3]. Different from prior work, the 207 sub-tasks in our problem are unknown. Our work designs 208 two latent tasks (learning the bounding-box and segmenta-209 tion layout) and a progressive learning strategy for effec-210 tive conditional image extrapolation. Although our sub-211 tasks share high-level similarity with text-to-image gener-212 ation approaches [11, 19], our progressive learning strategy 213 214 is different which separately trains each sub-task before the joint training. We have shown in Table 1 that it turns out 215

WACV 2021 Submission #685. CONFIDENTIAL REVIEW COPY. DO NOT DISTRIBUTE.



Figure 2. Framework of the proposed algorithm on progressive extrapolation. In stage I, we generate a bounding-box layout from the scene graph to roughly indicate the size and spatial location of each object. Then conditioned on the coarse bounding-box layout and the image patch, we learn to generate a semantic segmentation layout in stage II. Finally in stage III, we map the segmentation layout and the image patch to generate the extrapolated results. Details about the network architecture can be found in the supplementary material.

to be ineffective merely by incorporating these sub-tasks without progressive training (see "w/o pt" column). Similar definitions of sub-tasks are also found in text-to-image generation [11, 19]. Ours differ from them in two aspects: (i) we demonstrate it in the task of image extrapolation under the multi-modality conditioning; (ii) our progressive training contains an a joint training stage after separate training.

3. Proposed Method

Given an input image patch and a structured text represented as a scene graph, our goal is to extrapolate visual content beyond image boundaries that satisfies the conditions specified in the scene graph. We formulate this problem as a conditional image generation problem, where the conditions are the image patch, which specifies visual content in the known region of the target image, and the text (scene graph) which defines desired objects and their spatial relation to extrapolate for the unknown region.

Our model takes two inputs: an image patch \mathbf{z}_p and a structured text represented as a scene graph sg. We denote the input image patch as $\mathbf{z}_p \in \mathbb{R}^{h \times w \times 3}$ and the target image to generate as $\mathbf{x} \in \mathbb{R}^{H \times W \times 3}$, where h, H and w, Ware width and height of the images and h < H, w < W. We represent the text input as a scene graph [17]. Given a set of pre-specified object categories C and relationship categories \mathcal{R} , a scene graph is a tuple $\mathbf{sg} = (O, E)$ where $O = \{o_i | o_i \in C\}$ is a set of objects to extrapolate for the unknown region, and $E \subseteq O \times \mathcal{R} \times O$ is a set of directed edges specifying the relationship between objects. We focus on a common type of relationship in our problem, *i.e.*, the spatial relationship between objects which includes {left of, right of, above, below, inside, surrounding}.

Given an training example x drawn from the real distribution p_{real} and \mathbf{z}_p randomly cropped from x, our generation model learns a mapping function from \mathbf{z}_p and sg to the data space $\hat{\mathbf{x}} = G(\mathbf{z}_p, \mathbf{sg}; \theta_g) \in \mathbb{R}^{H \times W \times 3}$. In general, this learning process is self-supervised with a reconstruction loss L_{rec} and an adversarial loss L_{adv} [9]:

$$L_{total} = L_{rec} + \lambda L_{adv} = || \mathbf{x} - \hat{\mathbf{x}} ||_2^2 + \lambda L_{adv}, \quad (1)$$



Figure 3. Comparison of the standard training and the proposed progressive training.

where L_{adv} is computed by:

$$L_{adv} = \mathbb{E}_{\mathbf{x} \sim p_{real}}[\log D(\mathbf{x})] + \mathbb{E}_{\hat{\mathbf{x}} \sim p_{fake}}[\log(1 - D(\hat{\mathbf{x}}))],$$
(2)

where D is a discriminator to output a single scalar representing the probability of whether the x is real or not.

3.1. Overview

Directly optimizing Eq. (1) with deep generation networks (*e.g.*, [13, 37, 25, 6]) to translate unknown regions to plausible regions (*i.e.*, the standard training) only leads to blurry and less realistic outputs. Figure 3 shows an example training curve where the training loss (in blue) hardly decreases after a few epochs. The underlying reason is that using text to directly control RGB pixel generation is extremely difficult.

To address this issue, we design two latent sub-tasks that are closely related to our final generation task but are progressively easier to learn. Specifically, we train the generator progressively via three tasks where the output of a previous task is used in the next task. Let θ_g^* be the optimal parameter for our generator G and we find it by minimizing the total loss L_{total} over all training pairs of scene graphs and image patches:

$$L_{total} = L_{box}(\mathbf{sg}) + L_{seg}(\mathbf{x}_{bb}, \mathbf{z}_p) + L_{img}(\mathbf{x}_{seg}, \mathbf{z}_p),$$
(3)

where the losses L_{box} , L_{seg} , and L_{img} are used to estimate the negative log-likelihood for each generation of $p(x_{bb}|sg)$, $p(x_{seg}|x_{bb}, z_p)$, and $p(\hat{x}|x_{seg}, z_p)$, respectively.

325

326

327

328

329

330

331

332

333

334

335

336

337

338

339

340

341

342

343

344

345

346

347

348

349

350

351

352

353

354

355

356

357

358

359

360

361

362

363

364

365

366

367

368

369

370

371

372

373

With Eq. (3), the generation process is decomposed into three stages. First the bounding-box layout \mathbf{x}_{bb} is constructed from the scene graph. Then the segmentation layout \mathbf{x}_{seg} is created from the bounding-box and the input patch. Finally, the model generates the target image $\hat{\mathbf{x}}$ using the segmentation layout and the input patch.

Figure 2 illustrates the framework of our model. Our network first generates a *bounding-box layout* $\mathbf{x}_{bb} \in \mathbb{Z}^{|O| \times 4}$, a low-dimensional coordinate space for each object in the scene graph. Then the bounding-boxes are refined into a semantic *segmentation layout* ($\mathbf{x}_{seg} \in \mathbb{Z}^{H \times W \times 1}$), where each pixel is represented as a classification label of the object in O. The third stage maps the segmentation layout to the extrapolated RGB pixels $\hat{\mathbf{x}}$ via image-to-image translation. In the following, we describe the details of these stages.

3.2. Stage I: Bounding-box Layout Generation

The Stage I takes the scene graph as input and outputs a bounding-box spatial layout map. For the scene graph input, we use the graph convolution network (GCN) of [16] to transform object embeddings into the relationship-encoded representation. Given a graph with embeddings initialized at each node and edge, the GCN computes new embeddings for each node and edge through propagating information along edges of the graph. The edge embedding encodes the relationship between connected objects. The encoded object embeddings are then fed into a fully-connected network of three layers to predict the bounding-box coordinates $b\hat{b}_i$ for each object. Each box is represented as the top-left and bottom-right *x-y* coordinates. The loss in this stage is computed by the L_1 difference between ground-truth and predicted boxes:

$$L_{box} = \frac{1}{|O|} \sum_{i=1}^{|O|} || bb_i - \hat{bb_i} ||_1$$
(4)

where bb_i is the true bounding-box. Figure 4(a) shows two examples of the generated bounding-box layout for different scene graph inputs.

Note that sometimes scene graphs can be similar, *e.g.*, nearly all face images contain "eyes" and "nose" as the nodes. The lack of diversity makes it difficult to learn a good graph embedding. To address this issue, we augment the training data by randomly dropping some nodes out of the scene graph and meanwhile modifying the target image accordingly. We observe that the augmentation considerably enhances the controllability of the scene graph.

3.3. Stage II: Segmentation Layout Generation

The Stage II is responsible for transforming the coarse bounding-box layout into a segmentation layout conditioned on the image patch. As such, we need to accomplish three goals: (i) parse the known regions in the patch, (ii)



(b) Stage II: segmentation layout generation Figure 4. Examples of outputs of Stage I and II.

generate the segmentation layout for the unknown regions, and (iii) align the unknown and known regions.

The input to Stage II is the concatenated feature of the graph embedding from Stage I and the input image patch. We warp each node embedding in the scene graph using bilinear interpolation according to coordinates to compute a spatial vector that has the same shape as the input image. We use the network of [27] as the backbone architecture to infer the pixel-level object labels. Let $c_1, \ldots, c_N \in \{1, \ldots, |\mathcal{C}|\}$ be the target class labels for the pixels $1, \cdots, H \times W$ where $|\mathcal{C}|$ is the number of object categories and $N = H \times W$. This module is trained with pixel-wise multi-class cross-entropy loss:

$$L_{seg} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{c_i=1}^{|\mathcal{C}|} \omega_{c_i} \, y_{i,c_i} \log p_{i,c_i}, \tag{5}$$

where p_{i,c_i} is the predicted probability for pixel *i* of belonging to class c_i , and y_{i,c_i} is the binary label (0 or 1) indicating if class label c_i is a correct classification for pixel *i*. To handle the imbalanced classes (*e.g.*, "background" class is more common than "eye"), we use ω_{c_i} to downweigh the pixels from common classes.

Figure 4(b) shows two examples of the alignment between an existing eye in the given patch and the other eye generated outside boundaries. Given the same boundingbox layout, while the eyes of two conditional patches are at different height, our model is able to generate different segmentation layout that aligns with the input image patch well. This indicates that bounding-box layouts only impose soft constraints, and Stage II is able to recover the error from the Stage I output.

3.4. Stage III: Layout to Image Generation

Given the generated layout, the Stage III operates as a label-to-image mapping model in a way similar to imageto-image translation [13, 25]. Here we use a generic autoencoder with the instance normalization layer [35] for regularizing the network activations. The difference to imageto-image translation here is that our input is the concatenation of the segmentation layout and the input image patch.

429

430

436

437

438

458

459

460

461

462

463

464

465

466

467

468

469

470

471

472

473

474

475

486

487

488

489

490

491

492

493

494

495

496

497

498

499

500

501

502

503

504

505

506

507

508

509

510

511

512

513

514

515

516

517

518

519

520

521

522

523

524

525

526

527

528

529

530

531

532

533

534

535

536

537

538

539

To learn this model, we use the perceptual loss [15] and adversarial loss [1]:

$$L_{img} = \sum_{i=1}^{4} || \Phi_i(\mathbf{x}) - \Phi_i(\mathbf{\hat{x}}) ||_2^2 + L_{adv} , \qquad (6)$$

where x, \hat{x} are the ground truth and predicted image, and Φ_i is the pretrained VGG-19 [31] network up to the ReLU_*i*_1 layer.

442 **Remarks on training.** While all three tasks share the same goal of extrapolating valid objects that align well with the 443 given image patch, they are made increasingly difficult to 444 445 learn. For example, it is much easier to find the box lo-446 cations (in Stage I) than the RGB image (in Stage III) for 447 all objects to satisfy their relationship in the scene graph. 448 Likewise, it is a simpler task to align the input image patch 449 with the boxes than the final extrapolated content. There-450 fore, we first train each stage separately such that each stage 451 can focus on its own objective and learn a better initialized 452 model than random weights. However, the individual mod-453 els trained in these stages may cause errors when the in-454 termediate stage does not generate the precise layout. We 455 further jointly train all three models (from three stages) in 456 order to enforce the later stage to correct some inconsistent 457 outputs from the previous stage.

4. Experiments

We conduct experiments to validate the effectiveness of our model on two kinds of data, *i.e.*, real and synthetic data. For real data, we evaluate the proposed method on two types of object images of great interests, *i.e.*, face and human body. However, there always exist strong priors over certain object parts in real objects, which may degrade the controls from the scene graph. Hence, in order to better show the effectiveness of the scene graph, we design another experiment on a synthetic 2D shape dataset [43] where objects are randomly positioned without any prior. The source code and trained models will be made available to the public. More results and details can be found in the supplementary material.

4.1. Real Dataset

476 Dataset. As the first study of multi-modality conditional image extrapolation, we validate it on face and human 477 datasets of similar complexity as in contemporary extrap-478 479 olation works. The Helen dataset [18] consists of 2,330 480 face images with each face having 11 labels from [32] of main facial components. The Clothing Co-Parsing (CCP) 481 482 dataset [45] contains 1,004 images and corresponding label maps for 59 clothing items. Since the label classes are 483 484 highly unbalanced, we group similar labels (e.g., boots and 485 wedges are both treated as shoes) and create a super label set of 9 clothing items: {background, accessory, upper cloth, shoe, dress, hair, hat, pant, skin}. In this work, we choose not to employ complex scene datasets (*e.g.*, COCO [21], Cityscapes [7]) because extrapolating multiple complex objects is still too challenging for image extrapolation and no extrapolation works have ever been done on complex scenes. The experiments are conducted on these two datasets mainly because (i) face and human body are two types of object image of great interests, and (ii) compared with more complex scene datasets (*e.g.*, COCO [21], Cityscapes [7]) which only label the rough silhouette of objects, they contain more important detailed object parts.

The ground truth coordinates of the bounding-box of each label are computed by considering the smallest and largest coordinate of all pixels with the same label as the top left and the bottom right. Since both datasets do not provide annotated scene graphs, we construct the input scene graphs in a way similar to [16] from the ground truth position of each label in the image, with each label as the node and one of the six spatial relationships {left of, right of, above, below, inside, surrounding} as the edge.

During the training process, for each input image, we crop image patches of random size (around $15\%\sim25\%$ of the original image size) at random positions and train the network model to recover the original image. We fix the output size of extrapolated results which serves as a predefined canvas to restrict the scale of objects in the results. The extrapolated image sizes of face and human body in our work are 128×128 and 384×256 pixels respectively, which is $4\sim6$ times bigger than the size of input patches. For images in both datasets, we replace their original complex background, *i.e.* pixels of the label 0, with the clean white background to let the network focus on learning meaningful object parts.

Evaluated methods. Since there exist no exact extrapolation methods that can handle the multimodal input of image patch and scene graph, we compare with the following related work. The GMCNN [38] is the state-of-the-art image inpainting model. We adapt its original training objective from inpainting to outpainting pixels outside the patch boundary and keep the rest unchanged. As it does not support controls from the scene graph, we train the model only based on the image patch using their released code. The SRN [39] is the state-of-the-art model for image extrapolation. Similarly, the input to this model is an image patch only and we train the model using their code on both Helen and CCP dataset. The sg2im [16] is a closely-related prominent method to synthesize image from scene graph. As it does not take the image patch as input, we concatenate the image patch as additional input channels of their refinement network. We denote this variant as *sg2im_c* and use the code from [16] to retrain the model. In addition, by converting the scene graph to sentences, we also evaluate our method



9	U	35		Ŕ	1	Ì
---	---	----	--	---	---	---

Patch [38] [39] Ours Patch [38] [39] Ours Figure 6. Comparisons with non-text based inpainting/outpainting methods which directly generate the final output without taking the layout into account.

against the state-of-the-art text-driven I2I translation work **DMIT** [49] which has the same multimodal conditioning setting as our work, *i.e.*, using both text and image as conditions. Considering that **DMIT** is originally developed for unpaired data but our extrapolation task uses paired data, we add the perceptual loss in (6) into their objectives and use their code to retrain the model.

Qualitative Comparison. Figure 5 and 6 show the visual
comparisons between the proposed method and baselines,
where the former includes conditional extrapolation and the
latter contains unconditional extrapolation baselines. Given
the scene graph and conditional image patch in in Fig-

ure 5(a), our method generates more visually appealing and realistic results (e) than the scene graph based method (b) and text-based scheme (c). We also show the segmentation layout of second stage in our model in (d). Figure 6 shows that the inpainting and outpainting algorithms, which uses no text inputs, are missing the majority of pixels about detailed object parts (*e.g.*, the thin eyebrow and small head). Overall, our model generate sharper and more realistic results.

A unique property of conditional image extrapolation is being able to control image extrapolation with different text inputs. Figure 7 shows different extrapolated results of our model from the same image patch for different inputs. We randomly change the node or the relation of a given scene graph at a time. The results show that our extrapolation model follows the control signals specified in the scene graph and generate images that align well with the conditional image patch. These results tests our model is able to control extrapolation based on texts and images.

Quantitative results. We first evaluate the realism of the

WACV 2021 Submission #685	CONFIDENTIAL	. REVIEW COPY. D	OO NOT DISTRIBUTE.
---------------------------	---------------------	------------------	--------------------

Table 1. Quantitative evaluations on the Helen [18] and CCP [45] dataset.

		sg2im_c	DMIT [49]	GMCNN [38]	SRN [39]	Ours w/o pt	Ours
Halan	IS ↑	$1.42{\pm}0.08$	$1.58{\pm}0.14$	$1.40{\pm}0.11$	$1.48{\pm}0.12$	$1.45 {\pm} 0.11$	$1.82{\pm}0.16$
Helen	$FID\downarrow$	$70.02{\pm}1.53$	$56.84{\pm}1.31$	$71.28 {\pm} 1.22$	$67.69 {\pm} 1.63$	$62.34{\pm}1.27$	49.21±1.92
CCD	IS \uparrow	$3.01 {\pm} 0.34$	$3.36 {\pm} 0.26$	$3.24{\pm}0.29$	$3.37 {\pm} 0.27$	$3.14{\pm}0.31$	3.67±0.33
CCP	$FID\downarrow$	$119.77 {\pm} 0.19$	$83.46{\pm}1.32$	$95.88{\pm}1.49$	$86.85 {\pm} 1.22$	$97.24 {\pm} 0.78$	$\textbf{68.64}{\pm 0.17}$



extrapolated results, i.e. measuring how close the distribution of results is to that of the real data. We use two common metrics for general image generation tasks: Inception Score (IS) [28] and Fréchet Inception Distance (FID) [10]. We randomly crop patches on images in the test set and compute the metrics over 3,000 outputs of each model. Note that these metrics favor realistic and reasonable images completely neglecting the input texts (scene graph) and hence cannot evaluate the controllable setting. The evaluation results in Table 1 show that the proposed method achieves higher IS and lower FID scores across both datasets. Note that here we do not evaluate the layout generated in Stage I and II because there is no unique ground truth for an input patch under a controllable setting. Therefore we mainly focus on the evaluation of final results using the IS/FID metric and user studies, where humans can examine relevance between the final image and the given text.

We conduct user studies to analyse human perceptual
preference towards different methods. In addition to visual
quality, we also concern the relevance of generated images

Table 2. User preference towards different methods on real dataset.

		sg2im_c	DMIT [49]	Ours			
	Vote (%) ↑	8.35	17.43	74.22			
Table 3. Quantitative evaluations on the 2D shape dataset.							
	sg2in	n_c D	MIT [<mark>49</mark>]	Ours			

$\begin{array}{c} \text{IS}\uparrow\\ \text{FID}\downarrow \end{array}$	$\begin{array}{c} 1.31{\pm}0.24\\ 80.37{\pm}1.68\end{array}$	$\begin{array}{c} 1.66{\pm}0.19\\ 62.83{\pm}1.21\end{array}$	$\begin{array}{c} 1.75{\pm}0.22\\ 55.44{\pm}1.39\end{array}$

to the input scene graph. Thus here we only compare our model with *sg2im_c* and *DMIT* that are able to control the extrapolation by scene graph. We prepare extrapolated images for 20 (10 from Helen [18] and 10 from CCP [45]) pairs of scene graphs and patches. For each subject, we randomly select 15 pairs to evaluate and display the extrapolated results side-by-side in random order. Each subject is asked to vote the single best generated image that are (i) relevant to the given scene graph and (ii) realistic. We collect 300 votes from 20 participants who are not involved in the project. The user study is double-blind, *i.e.*, our results are shown unlabeled in randomized order and the identities of the participants are not disclosed. The user study results in Table 2 show that the proposed method receives the most votes, significantly higher than others. These results substantiate that our model is able to generate controllable image content that are more semantic relevant to the input scene graph.

Ablation study on progressive training. We compare with a variant of the proposed method in terms of training strategy. In contrast to the progressive training (pt) strategy used as default, we directly train all models of three stages from scratch and demoted this baseline as *Ours w/o pt*. For fair comparisons, we use the same set of cropped image patches in all methods. Results in Table 1 (see "w/o pt" column) show that it turns out to be ineffective merely by incorporating these sub-tasks without progressive training.

4.2. Synthetic Dataset

We observe that for real object data, there generally exist strong priors over certain object parts, *e.g.*, lips are always under noses in faces or sky is always above other objects. During training, network models will bias towards the prior

769

770

771

810

811

812





and ignore the input control signal. The prior inherently ex-772 ists in our natural world and every real dataset, simple or 773 complex, big or small. To demonstrate this, we use a sim-774 ple experiment by removing all relationships in the scene 775 graph. Figure 8(left) shows that with object nodes only, the 776 model is still able to generate a reasonable layout. How-777 ever, we do not want to completely lose the controls over 778 the extrapolation. As shown in Figure 5, we can still con-779 trol several items like the position of hair, pant or dress, and 780 with or without hat. 781

Therefore, to further validate the effective controls by the 782 scene graph, we conduct experiments on a synthetic dataset 783 of 2D shapes [43]. Each image in [43] contains three types 784 of objects (circles, squares, and triangles), which are ran-785 domly positioned to reduce the prior information (two ex-786 amples are presented on right of Figure 8). We show an 787 example of our extrapolation results in Figure 9. Quanti-788 tative evaluations listed in Table 3 show that our method 789 still obtains the best extrapolation results. Here we mainly 790 compare with sg2im_c and DMIT which also have the con-791 trollable setting. 792

793 While extrapolating 2D shape image is not of that great interest, below we mainly manipulate the scene graph to 794 795 show the bounding-box output of Stage I to illustrate the controllability. While positions are totally random between 796 797 objects, the scene graph is expected to be the only control signal. By controlling the scene graph, our model turns 798 out to be able to generate diverse bounding-box layouts 799 as shown in Figure 10, where each generated layout cor-800 rectly reflects the object and relationship information in the 801 scene graph. Note that each image in the original shape 802 803 dataset contains one circle, one square and one triangle only. 804 Our model generates more combinations of three categories (e.g., multiple circles) through controlling the scene graph. 805 This can be potentially used for graphic layout design to au-806 807 tomate the process of distributing different elements. One 808 interesting future direction is to add more detailed controls 809 (e.g., how far left or right, intersecting or tangent).



Figure 10. Flexible controls from scene graphs at Stage I. For better visualization, here we replace the bounding-boxes with objects.

From the experimental results on both real and synthetic datasets, we conclude that the controllability of scene graphs can be flexible but will be constrained, at least to some extent, by the data prior. It is also worth noting that although the scene graph provides control signals, we find it is insufficient to model rare objects or relationships. For example, it is unlikely to generate four left eyebrows if there are four left eyebrows in the scene graph. This should be expected because there exist no such cases in the training data.

5. Conclusion

In this work, we propose a generative network to extrapolate new content outside the image boundaries. Unlike image extrapolation, the studied extrapolation is controlled by a structured text (modeled as a scene graph) indicating what and where to generate for the unknown region. To realize controllable extrapolation, we decompose the learning process into three stages and introduced two important sub-tasks, of generating layouts from coarse to fine, to facilitate the training. Based on this multi-stage model, we use a curriculum learning strategy for effective model training. Both qualitative and quantitative results show that the proposed model performs favorably against the evaluated methods and is able to generate more controllable extrapolated results. Our future work includes modeling more complex scene images.

862

867

868

869

870

871

872

873

874

875

876

877

878

879

880

881

882

883

884

885

886

887

888

889

890

891

892

893

894

895

896

897

898

899

900

901

902

903

904

905

906

907

908

909

918

919

864 References

- Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. arXiv preprint arXiv:1701.07875, 2017.
- [2] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. In *ACM Transactions on Graphics*, volume 28, page 24, 2009. 2
- [3] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, 2009. 2
 - [4] Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. Image inpainting. In *SIGGRAPH*, 2000.2
 - [5] Haw-Shiuan Chang, Erik Learned-Miller, and Andrew Mc-Callum. Active bias: Training more accurate neural networks by emphasizing high variance samples. In *NIPS*, 2017. 2
 - [6] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *ICCV*, 2017. 1,
 3
- [7] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016. 5
 - [8] Alexei A Efros and William T Freeman. Image quilting for texture synthesis and transfer. In SIGGRAPH, 2001. 2
- [9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014. 2, 3
- [10] 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 *NeurIPS*, 2017. 7
- [11] Seunghoon Hong, Dingdong Yang, Jongwook Choi, and Honglak Lee. Inferring semantic layout for hierarchical textto-image synthesis. In *CVPR*, 2018. 2, 3
- [12] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and locally consistent image completion. ACM Transactions on Graphics, 36(4):107, 2017. 2
- [13] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 1, 3, 4
- [14] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *ICML*, 2018. 2
- [15] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual
 losses for real-time style transfer and super-resolution. In
 ECCV, 2016. 5
- 913 [16] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *CVPR*, 2018. 1, 2, 4, 5
- [17] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li,
 David Shamma, Michael Bernstein, and Li Fei-Fei. Image
 retrieval using scene graphs. In *CVPR*, 2015. 3

- [18] Vuong Le, Jonathan Brandt, Zhe Lin, Lubomir Bourdev, and Thomas S Huang. Interactive facial feature localization. In ECCV, 2012. 5, 7
- [19] Wenbo Li, Pengchuan Zhang, Lei Zhang, Qiuyuan Huang, Xiaodong He, Siwei Lyu, and Jianfeng Gao. Object-driven text-to-image synthesis via adversarial training. In *CVPR*, 2019. 2, 3
- [20] Yijun Li, Sifei Liu, Jimei Yang, and Ming-Hsuan Yang. Generative face completion. In *CVPR*, 2017. 2
- [21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In ECCV, 2014. 5
- [22] Guilin Liu, Fitsum A Reda, Kevin J Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *ECCV*, 2018. 2
- [23] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In ECCV, 2016. 1
- [24] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 2
- [25] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*, 2019. 1, 3, 4
- [26] Anastasia Pentina, Viktoriia Sharmanska, and Christoph H Lampert. Curriculum learning of multiple tasks. In CVPR, 2015. 2
- [27] Tobias Pohlen, Alexander Hermans, Markus Mathias, and Bastian Leibe. Full-resolution residual networks for semantic segmentation in street scenes. In *CVPR*, 2017. 4
- [28] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *NeurIPS*, 2016. 7
- [29] Qi Shan, Brian Curless, Yasutaka Furukawa, Carlos Hernandez, and Steven M Seitz. Photo uncrop. In *ECCV*, 2014. 1,
 2
- [30] Sahil Sharma, Ashutosh Jha, Parikshit Hegde, and Balaraman Ravindran. Learning to multi-task by active sampling. *arXiv preprint arXiv:1702.06053*, 2017. 2
- [31] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 5
- [32] Brandon M Smith, Li Zhang, Jonathan Brandt, Zhe Lin, and Jianchao Yang. Exemplar-based face parsing. In *CVPR*, 2013. 5
- [33] Jian Sun, Lu Yuan, Jiaya Jia, and Heung-Yeung Shum. Image completion with structure propagation. In ACM Transactions on Graphics, volume 24, pages 861–868, 2005. 2
- [34] Piotr Teterwak, Aaron Sarna, Dilip Krishnan, Aaron Maschinot, David Belanger, Ce Liu, and William T Freeman. Boundless: Generative adversarial networks for image extension. In *ICCV*, 2019. 2
- [35] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *CVPR*, 2017. 4

970

987

988

1026

1027

1028

1029

1030

1031

- [36] Miao Wang, Yukun Lai, Yuan Liang, Ralph Robert Martin, and Shi-Min Hu. Biggerpicture: data-driven image extrapolation using graph matching. *ACM Transactions on Graphics*, 33(6), 2014. 1, 2
- [37] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*, 2018. 1, 3
- [38] Yi Wang, Xin Tao, Xiaojuan Qi, Xiaoyong Shen, and Jiaya
 Jia. Image inpainting via generative multi-column convolutional neural networks. In *NeurIPS*, 2018. 2, 5, 6, 7
- 983[39] Yi Wang, Xin Tao, Xiaoyong Shen, and Jiaya Jia. Wide-
context semantic image extrapolation. In *CVPR*, 2019. 2, 5,
6, 79856, 7
 - [40] Sanghyun Woo, Dahun Kim, Donghyeon Cho, and In So Kweon. Linknet: Relational embedding for scene graph. In *NeurIPS*, 2018. 1
- [41] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei.
 Scene graph generation by iterative message passing. In *CVPR*, 2017. 1
- [42] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang,
 Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Finegrained text to image generation with attentional generative adversarial networks. In *CVPR*, 2018. 2
- [43] Tianfan Xue, Jiajun Wu, Katherine Bouman, and William Freeman. Visual dynamics: stochastic future generation via layered cross convolutional networks. In *NIPS*, 2016. 5, 8
- [44] Xinchen Yan, Jimei Yang, Kihyuk Sohn, and Honglak Lee.
 Attribute2image: Conditional image generation from visual attributes. In *ECCV*, 2016. 1
- [45] Wei Yang, Ping Luo, and Liang Lin. Clothing co-parsing by joint image segmentation and labeling. In *CVPR*, 2014. 5, 7
- [46] Zongxin Yang, Jian Dong, Ping Liu, Yi Yang, and Shuicheng
 Yan. Very long natural scenery image prediction by outpainting. In *ICCV*, 2019. 2
- [47] 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. 2
- [48] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and
 Thomas S Huang. Generative image inpainting with contextual attention. In *CVPR*, 2018. 2
- [49] Xiaoming Yu, Yuanqi Chen, Shan Liu, Thomas Li, and Ge
 Li. Multi-mapping image-to-image translation via learning
 disentanglement. In *NeurIPS*, 2019. 6, 7
- [50] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. Stack-gan++: Realistic image synthesis with stacked generative adversarial networks. arXiv preprint arXiv:1710.10916, 2017. 1, 2
 [1019] [
- [51] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017. 1, 2
- [52] Yinda Zhang, Jianxiong Xiao, James Hays, and Ping Tan.
 Framebreak: Dramatic image extrapolation by guided shiftmaps. In *CVPR*, 2013. 1, 2

- [53] Zizhao Zhang, Yuanpu Xie, and Lin Yang. Photographic text-to-image synthesis with a hierarchically-nested adversarial network. In *CVPR*, 2018. 2
- [54] Yang Zhou, Zhen Zhu, Xiang Bai, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. Non-stationary texture synthesis by adversarial expansion. In *SIGGRAPH*, 2018. 2

1078