Cost-Effective Language Driven Image Editing with LX-DRIM

Rodrigo Santos and **António Branco** and **João Silva**
University of Lisbon, Faculty of Sciences
NLX—Natural Language and Speech Group
Departamento de Informática, Faculdade de Ciências de Lisboa
Campo Grande, 1749-016 Lisboa, Portugal

Abstract

Cross-modal language and image processing is envisaged as a way to improve language understanding by resorting to visual grounding, but only recently, with the emergence of neural architectures specifically tailored to cope with both modalities, has it attracted increased attention and obtained promising results.

In this paper we address a cross-modal task of language-driven image design, in particular the task of altering a given image on the basis of language instructions. We also avoid the need for a specifically tailored architecture and resort instead to a general purpose model in the Transformer family.

Experiments with the resulting tool, LX-DRIM, show very encouraging results, confirming the viability of the approach for language-driven image design while keeping it affordable in terms of compute and data.

1 Introduction

The fields of image and language processing have mostly progressed independently of one other, each focusing on its own modality. Recently, though, there have been promising prospects for advancement in cross-modal processing. A major motivation for this has been the realization that the so-called grounding is necessary for progress in language understanding (Bisk et al., 2020), and a major enabling factor has been the emergence of underlying technology that can be successfully applied to both modalities and their cross-modal processing (Dosovitskiy et al., 2020; Ramesh et al., 2021; Wu et al., 2021; Radford et al., 2021).

In the image to language direction, there has been considerable progress in the task of image captioning, that is of generating a language description for an input image (Radford et al., 2021; Xu et al., 2015; Wu et al., 2017; Hossain et al., 2019), and the subsidiary task of image retrieval from a language description (Reed et al., 2016; Guo et al., 2018; Yu and Grauman, 2017; Kovashka et al., 2012); while in the language to image direction promising results have been obtained on the task of image generation from an input language description (Ramesh et al., 2021; Wu et al., 2021).

Conditional Generative models based on the Transformer architecture (Vaswani et al., 2017) became one of the mainstream approaches for virtually any language processing task (Radford et al., 2019; Brown et al., 2020; Devlin et al., 2018) due to their ability to cope with the intrinsically compositional nature of language and the meaning conveyed by contextualized expressions. Recently, these models have also shown promise for image processing tasks, namely in image generation (Ramesh et al., 2021; Wu et al., 2021), showcasing their capacity to handle multi-modal input, and how general purpose the Transformer architecture can be, coping also with data rooted in signals that are not linguistic in nature.

The DALL-E model (Ramesh et al., 2021) delivered promising results in such a task, by receiving a description in the form of a snippet of text (e.g. "a green clock in the form of an hexagon") and creating an image that humans recognize as one that could correspond to that input description. And its extension DALL-E 2 (Ramesh et al., 2022) undertakes also a more restricted task, where a specified subarea of the image is to be completed on the basis of the language description. These models achieve these results by leveraging massive quantities of data and compute that are hardly accessible to most research groups and organizations.

Adopting a distinct line of inquiry, in the present paper we aim at addressing a challenge of language driven image design, consisting of editing an image on the basis of language instructions to do so. Here the output image is conditioned not only on a text snippet but also on an input image, such that that image is appropriately altered taking into account the language input.

Proceedings of the 29th International Conference on Computational Linguistics, pages 31–43
October 12–17, 2022.

Figure 1: First (left to right): image with the caption "dark red pumps". Second: image generated (CIG model) with only the textual description in the caption of the first image. Third: outcome of the alteration of the second image (CIA model) with the instruction "are a darker red". Fourth: image retrieved from the database by using the second image for matching.

For example, given an image of a piece of furniture, the model is asked to change its color. And then possibly its height, shape, viewing perspective, or the direction of the light. This process should allow one to iteratively and interactively modify the design of some object without any specific image manipulation software, and with no knowledge of how to work with it.

This workflow can be exploited in a wide range of innovative applications, such as supporting a shopping assistant that progressively matches images altered by language instructions against current stock and suggests increasingly suitable products, among others examples.

Also concerned with addressing the issue of resource cost, in this paper we present exploratory research results on affordable Language Driven Image Design (LDID). The major contributions and findings of this study are: (i) a suitably instantiated GPT-2 (Radford et al., 2019) is an effective option to perform LDID; (ii) in what concerns the task of Conditional Image Generation, our approach offers a more streamlined setup than the one adopted in DALL-E; (iii) as a by-product of its ability for LDID, our model may usefully support the subsidiary task of image retrieval; and (iv) extending this set up with a pre-trained language model may improve the performance in some LDID tasks. This study resulted on the creation of a tool, LX-DRIM, for editing an image on the basis of language instructions.

The remainder of this document is structured as follows: Section 2 describes the neural model used in this study; Section 3 explains the experiments performed and introduces the data sets used; Section 4 presents the results obtained; Section 5 proceeds with error analysis; Section 6 discusses related work; and Section 7 closes the paper with concluding remarks.

2 Model

In looking for affordable LDID, we resorted to a GPT-2 small model (Radford et al., 2019), namely its current implementation from the transformers package of HuggingFace,[1] including their English pre-trained GPT-2 as well.[2]

GPT-2 has been successfully applied to virtually all language processing tasks. Given it was conceived for text, some adaptation is required in order for it to handle images. Interestingly, changes to the model architecture can be dispensed with, and the required adaptations can be restricted solely to the way the input data is pre-processed.

The minimal twist is to pass the images through a Vector-Quantized Variation Auto Encoder (VQ-VAE) that is both capable of describing an image with tokens according to an internal vocabulary of images and of constructing an image from those tokens (Ramesh et al., 2021).

Similarly to Variational Autoencoders, the main goal of VQ-VAEs is the encoding of an image into a vector, or group of vectors, that can then be decoded as closely as possible into the same original image. However, while in standard Variational Autoencoders, the latent space is continuous and is sampled from a Gaussian distribution, VQ-VAEs operate on a discrete latent space by maintaining a codebook. This codebook can then be used as vocabulary for text conditioned image generation.

Therefore, by passing an image through a VQ-VAE, one gets a sequence of tokens that represents the image. This sequence can be fed to a GPT-2 model like it is done with the sequence of tokens for language, given that the image tokens also have their own embedding in the embedding layer.

In this work we use the VQ-VAE from (Esser et al., 2021)[3], with a "vocabulary" for images of size 1024, which is added to the GPT-2 embedding map, and by means of which every image is represented.

With this extension to images in place, one can now proceed to train GPT-2 as it is done when it is applied solely to text, whereby given an input token it learns to predict the next one.

As training parameters for the GPT-2, we use a batch size of 6 with gradient accumulation of 16, meaning that at each step our model back-propagates with 96 training instances. We evaluate

[1] https://huggingface.co/docs/transformers/index
[2] https://huggingface.co/gpt2
[3] https://github.com/CompVis/taming-transformers

on the development set every 250 steps, and stop training when the development set loss does not decrease from its lowest point after 5 evaluations.

After the training of the GPT-2 model, we optionally rank its outputs using CLIP[4] over the various images from the same input. After using two separate encoders, for image and for text, CLIP maps their encoding vectors into a common embedding so that a caption and its respective image end up with the same representation (Radford et al., 2021). CLIP can thus support the ranking of images generated from a caption given the encoded image that is closer (in vector space) to the encoded caption is the one more closely described by the caption.

3 Experiments

With this model in place, the following experiments were undertaken:[5] (i) a warm up experience, aimed at assessing the capability of the model for Conditional Image Generation (CIG)—generating an image from a text snippet describing it; (ii) the central experiment of interest here, aimed at assessing how well the model is able to perform Conditional Image Alteration (CIA)—generating an image both from another image and from a text snippet describing how the later should be altered; and, in addition, (iii) a comparison between the model and a variant obtained by extending it with a language pre-training phase.

3.1 Data sets

We resorted to the two data sets developed by (Guo et al., 2018)[6] for their research on image retrieval, which we re-purposed for the tasks of interest here, which differ from that original image retrieval task.

These data were developed through crowdsourcing with Amazon Turk and include: (i) a dataset of images of women shoes and respective captions, re-purposed here for the CIG task; and (ii) a dataset where each instance contains a source image of a shoe, a target image of another shoe, and a short textual description of how the source image relates to the target one, re-purposed here for the CIA task. Figure 2 shows an example from each data set.

The data set for CIG has 3600 examples. We randomly shuffled it and produced a 80/10/10 split, taking 2880 examples for training, 360 for development and the remaining 360 for testing. The data set for CIA, in turn, has 10750 examples, and it was also shuffled and submitted to a 80/10/10 split, with a 8600 example set for training, 1075 for development and 1075 for testing.

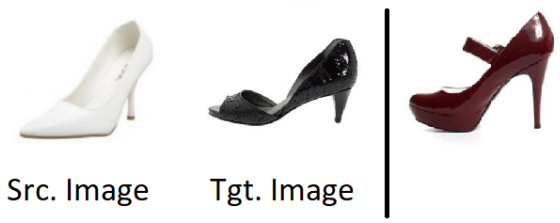

Figure 2: Left image: example in the CIA dataset, where the pair of images are associated to this textual instruction for the source image to be altered into the target image: “are black with a thicker heel”. Right image: example in the CIG dataset, associated to the caption “dark red platform high heels with a strap”.

All images in these data sets are augmented via several transformations: (i) images are flipped horizontally with a 50% chance; (ii) rotated between 0º and 20º clockwise or anticlockwise; (iii) distorted in order to simulate different perspectives with a 50% chance; (iv) their sharpness increased by a factor of 2 with a 50% chance; and finally (v) their contrast is maximized with a 50% chance.

3.2 Input representation

3.2.1 Conditional Image Generation

For each instance in the CIG data set, 194 input tokens were used: 128 text tokens, with the image caption; followed by a delimiter token (<I>) indicating where the image begins; followed by the 64 tokens output by the VAE, which represent the image; and finally, another <I> token indicating the end of the image.

During preliminary experimentation, we varied the number of tokens that represent the image and observed that using more tokens created a higher resolution image at the cost of the image being less precise. We empirically found that using 64 tokens to represent the image led to a good trade-off between image quality and precision.

Also in preliminary experimentation, while experimenting with other data sets not used in this study, another finding was that using images with white backgrounds helped the model to focus on the main object, being difficult for the model to precisely detect the object in question when the image had a noisier background.

[4] https://github.com/openai/CLIP

[5] Materials for the reproduction of the results reported here are available at https://github.com/nlx-group/LX-DRIM.

[6] https://github.com/XiaoxiaoGuo/fashion-retrieval

3.2.2 Conditional Image Alteration

For each instance in the CIA data set, 259 tokens were used: 128 text tokens with the request for alteration; a <I> token marking the beginning of the source image; 64 image tokens from the source image; a <I> token marking both the end of the source image and the beginning of the target image; another 64 tokens from the target image; and finally, a <I> token marking the end of the target image.

Our initial approach was to provide the source image first, followed then by the textual alteration. However, the resulting model had worse performance than the one with the text in the first (leftmost) place, as described above. This is possibly due to the fact that, by having the textual tokens first, the model can more easily learn the point from which no more textual tokens occur—after the first <I>—and after that point can attribute low probabilities to textual tokens and focus solely on generating image tokens.

3.2.3 Impact of CLIP

The notion of prompt engineering has emerged in papers like the ones regarding GPT-2 (Radford et al., 2019) or GPT-3 (Brown et al., 2020), and also DALL-E (Ramesh et al., 2021) or CLIP (Radford et al., 2021). This concerns how the textual input is given to the model and how the user can condition it to deliver the desired result.

Similarly to what is reported in those papers, the performance of our CIA model improves when the description of the object in the source image is included in the alteration text, instead of this text only stating the alteration to perform—e.g. “high heels are a darker tone” vs. “are a darker tone”. This can be partly attributed to the fact that the model gets a confirmation of what image to generate ("high heels" vs. "rain boots"). We use this approach to help CLIP rank the generated images, by prefixing the textual input with the expression denoting the type of object of the source image.

While the type of object of the source image may not always be the same as that of the target image, in general a prompt prepared this way improves the performance when CLIP is used for ranking.

4 Results

The evaluation of a generative task (e.g. summarization, etc.), where typically there can be more than one output that is acceptable as correct, tends to be a problematic endeavour. While one could try to compare to a gold standard in order to perform an automatic evaluation, small differences (of equally acceptable outputs) to the gold example inevitably makes most such metrics, like accuracy, etc., useless, leaving only some kind of distance metric to be resorted to.

In contrast to text processing, this problem tends to be further aggravated for images, as metrics that are used to evaluate textual generative tasks, like BLEU (Papineni et al., 2002) or METEOR (Banerjee and Lavie, 2005), work by being able to refer to some parts that are well defined substructures in an expression (e.g. words), but for images there are no clear substructures that can be resorted to, and in most cases these distance metrics work only at the pixel level.

4.1 Distance metrics

Given these considerations, we resorted to four distance metrics, two of which are hash functions:[7] Average hash (A. Hash), which takes the shape into consideration but compares the images in gray scale; Color hash (C. Hash), similar to A. Hash but taking color into consideration; Mean Square Error (MSE), the most rudimentary metric used, which focuses on the distance between pixels; and Structural Similarity Index Measure (SSIM) (Wang et al., 2004), one of the most used metric for image comparison, which extracts luminance, contrast and structure to compare two images. For the first three, lower scores are better, while for SSIM higher scores are better.

The results obtained with these automatic metrics will help to converge onto the more favorable settings for the model whose performance will eventually be submitted to the manual evaluation.

4.2 Conditional Image Generation

Table 1 presents the results obtained for CIG, where images are generated from text descriptions.[8] All evaluation scores were obtained as the mean score of the top four ranked images, with the exception of the last line (as only one image was available). The data for this task are available at https://github.com/nlx-group/LX-DRIM, which also include the images generated.

The best results under each metric concentrate in the middle of the table, when CLIP is fed with

[7] https://pypi.org/project/ImageHash/

[8] Running on an NVIDIA 2080 RTX 8G, CIG models were trained in 7 and 3 GPU hours, with and without language pre-training respectively.

N. Examples	A. Hash	C. Hash	MSE	SSIM	A. Hash	C. Hash	MSE	SSIM
	Without textual pre-training				With textual pre-training			
32	13.553	**4.6313**	0.0868	0.5587	13.480	4.6458	0.0832	0.5658
16	**13.510**	4.6326	0.0859	0.5614	13.434	4.6340	0.0824	0.5691
8	13.594	4.6681	**0.0850**	**0.5638**	**13.326**	**4.6285**	**0.0810**	**0.5741**
4	18.826	5.0792	0.0859	0.5290	19.441	5.0451	0.0830	0.5226
1	33.575	6.0417	0.0916	0.4072	35.875	6.2944	0.0901	0.3704

Table 1: Evaluation of CIG with the averaged scores of top-4 images, with (right half) and without (left) textual pre-training, with four image distance metrics (columns): Average Hash, Color Hash, Mean Square Error (lower is better), and Structural Similarity Index Measure (higher is better). The first column indicates the number of generated images (8, 16 and 32) given to CLIP.

eight examples. This indicates that using CLIP improves performance only to a certain point, after which increasing the number of examples given to it induces a detrimental effect.

With only one image generated, the model has the worst performance as there is no ranking to exclude the worst images. However, with four images generated (which also do not pass through CLIP), there are better scores than with only one, indicating that the model is more prone to creating more precise images than imprecise ones, and that by having multiple images the error is averaged out.

Considering the best scores with each metric, the models pre-trained with language data (right half of the table) have better performance than those that do not have such pre-training (left half). This may hint at that language pre-training is still relevant when there are images also in the fine-tuning phase.

4.3 Conditional Image Alteration

Table 2 presents the results for CIA, where images are generated both from other images and from text describing the alterations requested.[9]

The scores for the contribution of CLIP here are less consistently aligned with each other. Like in CIG, in general, a lower number of examples fed into CLIP seems to lead to better results.

In fact, with the SSIM metric, the best results are obtained with CLIP being fed with the lower number (8) of examples. However, for the hash metrics, it is hard to find such clear trend, other than that CLIP supports the best scores—in many setups with less examples, but in a few others with more. And while lower number of examples fed into CLIP also leads to better results with the MSE metric, their best results, in turn, are obtained without CLIP.

Additionally, considering the best scores with each metric, in some metrics one gets better results with textual pre-training, while with others is the other way around. These results are thus inconclusive with regards whether performance improves with or without textual pre-training for CIA.

4.4 Calibration

As an opportunistic extension or application of our model, its conditional image editing capability can easily support an image retrieval system. This can be achieved by measuring the distance, from the image generated for the input description, to every image in a database and retrieve the one that is found to be the most similar.[10]

While the performance of this kind of approach is likely inferior when compared to the feature-based methodology typically used in image retrieval systems, it is still worth experimenting with it. This will have the virtue of helping to assess the reliability of each one of the four evaluation metrics we have been using: given every metric is agnostic to the dataset, the domain or the model, and with no possible bias sensitive to any of them, the one with more matches to the gold counterparts will turn out to be the best to be used to evaluate image design tasks.

We evaluate the CIG model, with language pre-training, with 8 images generated (and filtered to 4 by CLIP), for its retrieval accuracy within the top 50, 10, 5 and 1 images retrieved, resorting to

[9]Running on an NVIDIA Titan RTX 24G, CIA models were trained in 17 and 7 GPU hours, with and without language pre-training respectively. Model inference (image generation) took less than a second.

[10]It is worth noting again that the data set we are using (Guo et al., 2018) was originally developed to support a image retrieval task, which the authors addressed by means of a complex system that takes into account the user feedback so that at each turn the system tends to get closer to the correct image to be retrieved.

N. Examples	A. Hash	C. Hash	MSE	SSIM	A. Hash	C. Hash	MSE	SSIM
	Without textual pre-training				With textual pre-training			
32	14.272	**4.2679**	0.1103	0.5339	14.583	4.7551	0.1109	0.5352
16	**13.952**	4.2842	0.1076	0.5399	**14.381**	4.7409	0.1100	0.5401
8	14.431	4.6902	0.1074	**0.5464**	14.431	**4.6902**	0.1074	**0.5464**
4	17.633	4.3937	**0.1041**	0.5459	20.102	5.1612	0.1040	0.4976
1	27.836	4.8112	0.1049	0.5173	34.122	6.2688	**0.0967**	0.3650

Table 2: Evaluation of CIA.

N. Retrieved	A. Hash	C. Hash	MSE	SSIM
50	33.61%	30.28%	46.67%	9.17%
10	10.00%	11.11%	15.00%	1.94%
5	5.56%	6.39%	8.33%	1.39%
1	1.67%	1.39%	1.67%	0.28%

Table 3: Accuracy of retrieving images with images generated from their captions by the CIG model where the retrieval is based in each of the four distance metrics (columns), for top-k retrieved images (first column).

the 360 examples in the test set. The respective evaluation scores are displayed in Table 3.

These results on image retrieval are low, being, nevertheless, above the random baseline (1/360 or 0.27% for 1 image retrieved). We tend to attribute these low results mainly to the nature of the data set as most images are very similar to each other—more on this below, in Section 5.

Nonetheless, the important take away sought for is the comparison between the four metrics, and their calibration to serve as evaluation metrics for our tasks of interest. Whereas MSE is the metric with higher scores at all settings considered (i.e. each line in the table), SSIM gets the lower scores, practically at random performance, being only 0.01% above it when one image is retrieved. Hash metrics, in turn, perform practically on a par with each other, with A. Hash performing slightly above C. Hash for 1 and 50 retrieved images, and C. Hash performing above A. Hash for 5 and 10 images. Accordingly, these results indicate that MSE could be considered as a more reliable distance metric than the other three.

4.5 Evaluation

Taking these preparatory findings into account, the model was evaluated in the task of interest here, CIA, under what appears as its most suitable settings following MSE scoring, with one example generated and language pre-training.

Two test sets were gathered, each with 25 randomly selected examples. Test set A (cf. Appendix A.1) consisted of triples with, from left to right in each line, source image, image produced by the model, and the alteration instruction. In test set B (cf. Appendix A.2), the examples consisted of 4-ary tuples with, from left to right, the source image, the gold target image, the image output by our model, and the instruction for alteration.

Six independent and voluntary evaluators were assigned the following task: given the original image on the left and the alteration instruction, classify how much the image on the right is a satisfactory result with a score from $\{1, 2, 3, 4\}$, where 4 indicates that it is fully satisfactory. They ran the evaluation over the entire test set A first, and then over the test B. To avoid eventual prejudice and respective bias, they were not told that images were generated by computer.

The averaged mean ratings of the evaluators was 2.37 (s.d. 0.11) with test set A. With test set B, the perceived quality slightly lowered to 2.26 (s.d. 0.36), showing that evaluators' rating tended to be pulled down by their seeing a result deemed as fully satisfactory side by side to the one under evaluation.

To evaluate also the CIG task, as DALL-E is not available, we resorted to its HuggingFace smaller version, DALL-E mini,[11] to generate images from 25 randomly selected captions in our data set (cf. Appendix B). Our model was also run on these captions. Following the same comparative evaluation approach used for CIG in DALL-E, in a best-of-five vote, the images generated by our model were always chosen as the most realistic and as best matching the caption. The images generated by the other system happen to be scrambled pieces of disparate objects.

When compared to our model DALL-E mini has

[11] https://huggingface.co/flax-community/dalle-mini

3 times more parameters (400 million vs 124 million) and was trained on 5000 times more images (15 million vs 2880).

5 Error analysis

To help in error analysis, difficult cases are exemplified in Figure 1. The two leftmost shoes are, respectively, the target image and the (CIG) generated image with the description "dark red pumps".

Both shoes are quite similar in terms of shape, but their color is different. This is a good illustration that color saturation and lightness are subjective and hard to transmit via text. In the target image (1st column), the desired dark red is almost black, and the image generated (CIG) from "dark red pumps" (2nd column) is lighter.

Interestingly, even the tentative correction (CIA) of this image with the instruction "are a darker red" still does not produce an image (3rd column) that is not as dark as in the first column.

Though image retrieval is not a central task of interest in this paper, it is worth noting that this may be even more serious for image retrieval as slight changes in saturation and lightness can make the system choose a different image: When trying to retrieve an image from the database, using the generated image (2nd column), the image that is retrieved is the one at the fourth column.

Further difficult examples, generated by the CIA model, are shown in Figure 3.

One problem illustrated there concerns image clarity. Even though some images (see 1st column) are correct, they have some fuzzy details. This is likely due to the reduced volume of the training data set. However, as already mentioned, in order to have images with higher resolution given a data set of this size, one would have to sacrifice image relevance and precision.

Another problem arises when the target image is very different from the source image (see 2nd column). In such cases, the model is basically asked to create a quite different object, for which the small size of the data set provided limited evidence.

Additional problems occur when the images to be generated are too similar to the source image (see 3rd column), or the generated images are too similar to each other (see 3rd and 4th images in the 1st column). While not necessarily a problem for the overall quality of the output, the first kind of cases becomes an issue for evaluation, as generated images may be more similar to the source image than to the target one. As for the second kind of cases, when the generated images are similar to one another, it may become a problem if object design is the intended use for the tool, and not just image alteration.

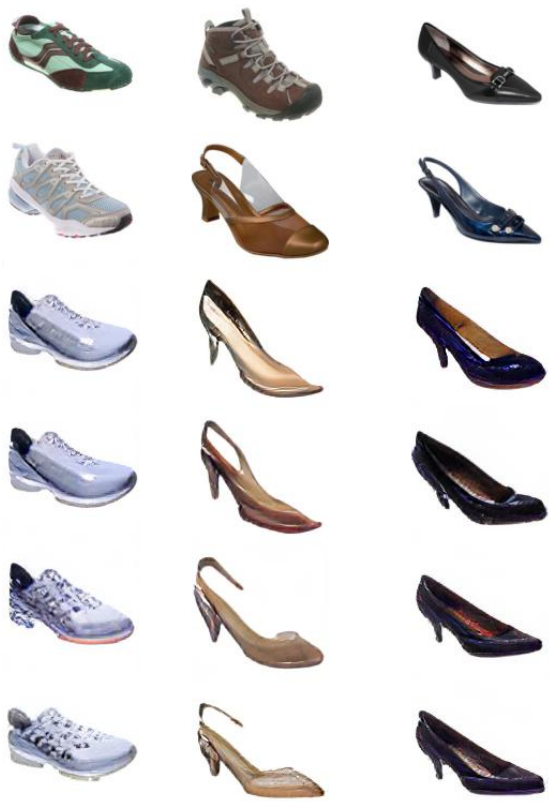

Figure 3: Examples of CIA for error analysis. First row: source images. Second row: target images. Remaining rows: top four generated images. Textual instructions for image alteration in left column: "athletic shoes are blue and silver"; middle column: "athletic shoes are bronze-colored slingbacks"; right column: "pumps are blue".

To address these issues, further techniques to enhance image diversity should be explored in future work, so that the model can suggest a more varied set of images to the user.

6 Related Work

A promising application of deep learning to image generation was presented in (Goodfellow et al., 2014), with a Generative Adversarial Network (GAN), a forerunner of a research line continued in (Xu et al., 2017), (Zhu et al., 2019), (Tao et al., 2021), a.o. A two part network containing a generator and a discriminator was proposed: The generator tries to create fake yet as realist as possible images, while the discriminator tries to distinguish

the fake images produced by the generator from real ones.

Despite this early success being attributed also to the use of Convolution Neural Networks (CNN) (LeCun et al., 1989), the concept of GAN can be used with other deep learning approaches. Such is the case of the more recent work in (Jiang et al., 2021b), where two Transformer models (Vaswani et al., 2017) are used as a discriminator and a generator respectively. With no convolution at its core, they achieve competitive scores when compared to their CNN counterparts.

Transformers gained their notoriety with their success in languages processing tasks of all kinds, and recently they have been applied to other data modalities. Relevant models that use Transformers for Image Generation from captions are DALL-E (Ramesh et al., 2021), and NUWA (Wu et al., 2021). The major difference between them is that NUWA also uses video while DALL-E works only with pictures, and that NUWA uses a different type of attention mechanism, 3D Nearby Attention.

The approach proposed in (Galatolo et al., 2021) also achieves promising results in image generation with a pre-trained Transformer CLIP (Radford et al., 2021), only by training a genetic algorithm.

More recently DALL-E 2 (Ramesh et al., 2022) improves upon its predecessor by incorporating the CLIP model for image and caption representation, and through the use of a diffusion model for image generation (Dhariwal and Nichol, 2021).

The architecture adopted in our model is similar to the backbone architecture on which the implementation of DALL-E is based. Our model is different from DALL-E, however, in not having any specific optimization performed on the base Transformer, like it was done to set up DALL-E, and in being of a more reduced size (124M vs. 12B parameters). Our system also differs in that it is geared for a task other than the Conditional Image Generation one, of DALL-E, namely the task of Conditional Image Alteration. It happens also that it was trained in a much smaller amount of data (10750 vs. 250 million examples).

Also, related to our research topic, (Cheng et al., 2020) tackles the same task, though by means of a Generator/Discriminator architecture, with data that while similar to ours is not the same. To the best of our knowledge, that dataset is not publicly available, so no comparison was possible. (Jiang et al., 2021a) also work with language guided image edition, with different datasets that do not tackle the problem of object shape manipulation.

Work on image editing without language guidance can be found in the work of (Zhu et al., 2020; Zhuang et al., 2021), on different datasets.

The research presented here appears as a more streamlined approach for the tasks involved in Language Driven Image Design since most of the work is performed with a common decoder-only architecture, in the form of a GPT-2 small model. This is a generalist architecture that can be adapted for other tasks, as it was the case here with the CIG task, or any other task that can be represented by a sequence (text, audio, image, etc.).

7 Conclusion

The present study explored Conditional Generative models for Language Driven Image Design, by means of an affordable GPT-2 instantiation with only 124M parameters. The central task of interest here was Conditional Image Alteration, consisting of generating a new image given a source image and a textual instruction for its alteration, on which the proposed LX-DRIM application showed a performance rated at 2.37 (in 1–5) by manual evaluators.

Resorting to the same data set, the task of Conditional Image Generation, consisting of generating an image given a textual description, was also experimented with. Very encouraging results were also obtained, specially taking into account that the data set used here was several orders of magnitude smaller than the one that has been used in the literature for this task.

In addition, we found also that as by-product of its cross-modal processing ability, our model may usefully support the subsidiary task of image retrieval through the use of its generated images.

Empirical experimentation obtained very encouraging results and demonstrated that the proposed approach can support an effective solution to Language Driven Image Design and represents a promising research path whose potential is worth being further exploited.

The present study focuses on changing a single object in the image, rather than multiple objects in a scene. Future work the task of scene manipulation (El-Nouby et al., 2019; Zhang et al., 2021) should be investigated by exploiting the approach developed here with single object manipulation.

Acknowledgments

The research reported here was supported partially by PORTULAN CLARIN—Research Infrastructure for the Science and Technology of Language, funded by Lisboa 2020, Alentejo 2020 and FCT—Fundação para a Ciência e Tecnologia under the grant PINFRA/22117/2016.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, et al. 2020. Experience grounds language. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8718–8735.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Yu Cheng, Zhe Gan, Yitong Li, Jingjing Liu, and Jianfeng Gao. 2020. Sequential attention gan for interactive image editing. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4383–4391.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Prafulla Dhariwal and Alexander Nichol. 2021. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Alaaeldin El-Nouby, Shikhar Sharma, Hannes Schulz, Devon Hjelm, Layla El Asri, Samira Ebrahimi Kahou, Yoshua Bengio, and Graham W Taylor. 2019. Tell, draw, and repeat: Generating and modifying images based on continual linguistic instruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10304–10312.

Patrick Esser, Robin Rombach, and Bjorn Ommer. 2021. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12873–12883.

Federico A Galatolo, Mario GCA Cimino, and Gigliola Vaglini. 2021. Generating images from caption and vice versa via CLIP-guided generative latent space search. *arXiv preprint arXiv:2102.01645*.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.

Xiaoxiao Guo, Hui Wu, Yu Cheng, Steven Rennie, Gerald Tesauro, and Rogerio Schmidt Feris. 2018. Dialog-based interactive image retrieval. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 676–686.

MD. Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. 2019. A comprehensive survey of deep learning for image captioning. *ACM Comput. Surv.*, 51(6).

Wentao Jiang, Ning Xu, Jiayun Wang, Chen Gao, Jing Shi, Zhe Lin, and Si Liu. 2021a. Language-guided global image editing via cross-modal cyclic mechanism. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2115–2124.

Yifan Jiang, Shiyu Chang, and Zhangyang Wang. 2021b. Transgan: Two transformers can make one strong GAN.

Adriana Kovashka, Devi Parikh, and Kristen Grauman. 2012. Whittlesearch: Image search with relative attribute feedback. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2973–2980. IEEE.

Yann LeCun, Bernhard Boser, John Denker, Donnie Henderson, Richard Howard, Wayne Hubbard, and Lawrence Jackel. 1989. Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*, 2.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. Zero-shot text-to-image generation. *arXiv preprint arXiv:2102.12092*.

Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. 2016. Generative adversarial text to image synthesis. In *International Conference on Machine Learning*, pages 1060–1069. PMLR.

Ming Tao, Hao Tang, Songsong Wu, Nicu Sebe, Xiao-Yuan Jing, Fei Wu, and Bingkun Bao. 2021. Df-gan: Deep fusion generative adversarial networks for text-to-image synthesis.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612.

Chenfei Wu, Jian Liang, Lei Ji, Fan Yang, Yuejian Fang, Daxin Jiang, and Nan Duan. 2021. NÜWA: Visual synthesis pre-training for neural visual world creation. *arXiv preprint arXiv:2111.12417*.

Qi Wu, Chunhua Shen, Peng Wang, Anthony Dick, and Anton Van Den Hengel. 2017. Image captioning and visual question answering based on attributes and external knowledge. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1367–1381.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR.

Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. 2017. Attngan: Fine-grained text to image generation with attentional generative adversarial networks.

Aron Yu and Kristen Grauman. 2017. Fine-grained comparisons with attributes. In *Visual Attributes*, pages 119–154. Springer.

Tianhao Zhang, Hung-Yu Tseng, Lu Jiang, Weilong Yang, Honglak Lee, and Irfan Essa. 2021. Text as neural operator: Image manipulation by text instruction. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 1893–1902.

Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. 2020. In-domain gan inversion for real image editing. In *European conference on computer vision*, pages 592–608. Springer.

Minfeng Zhu, Pingbo Pan, Wei Chen, and Yi Yang. 2019. Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5795–5803.

Peiye Zhuang, Oluwasanmi Koyejo, and Alexander G Schwing. 2021. Enjoy your editing: Controllable gans for image editing via latent space navigation. *arXiv preprint arXiv:2102.01187*.

A CIA Manual Evaluation Sheet

A.1 TEST A

First page of the test set A. Remaining pages can be consulted at https://github.com/nlx-group/LX-DRIM. From left to right: source image, generated image, and text snippet with alteration request.

		Are a lime green color	
		Are shiny grey clogs	
		Is white	
		Are metallic silver	
		Black flats	
		Is busier with contrasting panels and strap	

A.2 TEST B

First page of the test set B. Remaining pages can be consulted at https://github.com/nlx-group/LX-DRIM. From left to right: source image, target gold image, generated image, text snippet with alteration requested.

	Is plum, not black	
	Is gold	
	Is busier with laces and zipper with rugged sole	
	Have no heels	
	White sneakers with blue trim	
	Are yellow	
	Are blue and silver	

B CIG Manual Evaluation Sheet

First page of the CIG test set. Other pages can be consulted at https://github.com/nlx-group/LX-DRIM. From left to right: image caption, image generated by our system, image generated by DALL-E Mini.

ballet flats		
beige sneakers		
black flats with design		
black low heel motorcycle boot		
black mid-heeled long-on-the-leg boots