# Generating Images Instead of Retrieving Them: Relevance Feedback on Generative Adversarial Networks

Antti Ukkonen, Pyry Joona, Tuukka Ruotsalo
Department of Computer Science, University of Helsinki and Helsinki Institute for Information Technology HIIT
Helsinki, Finland
first.last@helsinki.fi

## ABSTRACT

Finding images matching a user's intention has been largely based on matching a representation of the user's information needs with an existing collection of images. For example, using an example image or a written query to express the information need and retrieving images that share similarities with the query or example image. However, such an approach is limited to retrieving only images that already exist in the underlying collection. Here, we present a methodology for generating images matching the user intention instead of retrieving them. The methodology utilizes a relevance feedback loop between a user and generative adversarial neural networks (GANs). GANs can generate novel photorealistic images which are initially not present in the underlying collection, but generated in response to user feedback. We report experiments (N=29) where participants generate images using four different domains and various search goals with textual and image targets. The results show that the generated images match the tasks and outperform images selected as baselines from a fixed image collection. Our results demonstrate that generating new information can be more useful for users than retrieving it from a collection of existing information.

## CCS CONCEPTS

• **Information systems → Users and interactive retrieval**.

## KEYWORDS

Image search; GAN; Relevance feedback

**ACM Reference Format:**
Antti Ukkonen, Pyry Joona, Tuukka Ruotsalo. 2020. Generating Images Instead of Retrieving Them: Relevance Feedback on Generative Adversarial Networks. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20), July 25–30, 2020, Virtual Event, China.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3397271.3401129


*SIGIR '20, July 25–30, 2020, Virtual Event, China*

ACM ISBN 978-1-4503-8016-4/20/07…$15.00
https://doi.org/10.1145/3397271.3401129

## 1 INTRODUCTION

Image search has become a popular activity within web search engines, now accounting for a large proportion of all search activity on the web. Despite its popularity, designing image search systems has turned out to be challenging because expressions of information needs toward images can be difficult to construct [32]. Most of the existing web search engines allow users to formulate queries as keywords or phrases. However, expressing image-related information needs and intents as keywords can be difficult when users can imagine what they want but are unable to express this in precise words [4, 16, 25, 32]

This is partly because esthetic properties, compositional details, and unmanifested needs toward the images arise while users interactively search images, making the search process harder than simply matching a query. Users may also have diverse information needs and intentions even for the same exact queries [14, 29]. For example, a user searching for a "red sports car" could conduct the search in order to include a conventional car image in a presentation, while another user might run the exact same query with an intent to include several unconventional images of red cars on a mood board for inspiration within a design assignment. Therefore, image search is prone to vaguely specified and even unexpressed needs, leading to an extensive interactive process where adjustments and refinements by the user are necessary to explore the image space.

Despite these challenges, it has been recognized that users often have in their mind an "ideal" target image, or at least an idea of what the target image should look like. Searching that image via an image search tool, however, turns out to be difficult and may lead only to approximate results [18]. This problem roots both to availability of the images and the gap between the expressions of users' needs and the computational representation of the images. Even though there are billions of images indexed from the web, these cover only a small portion of all images that a user could imagine. On the other hand, even if an "ideal" image would exist in the index, communicating the specifics of that image to the search engine is often beyond the functionalities of current image search systems.

Previous work has approached the problem by providing interactive relevance feedback techniques to assist users in searching images more efficiently [20, 22, 34]. These vary from query support and augmentation [20] to semantic understanding of the content [6, 30], and interactive approaches, such as sketching-based search [2].

While partly effective, these techniques are based on retrieving existing images from a finite collection and thus cannot answer

users' information needs when an exact image matching the user's "ideal" image does not exist in the indexed collection.

With this motivation, we ask the following questions: What if we were not constrained by the images that already exist in our collection? What if information retrieval systems did not only *retrieve* relevant images but would allow to interactively *generate* completely new images that match a user's information needs?

We present a methodology based on interactive relevance feedback and generative adversarial networks (GANs) [8, 12] that generate images matching a user's intentions rather than just retrieving best matches from a fixed collection. We demonstrate an interactive system implementation of the methodology and report a user experiment (N=29) with the system indicating the effectiveness of our approach in practical image generation. Our results show performance improvements over image retrieval in various image subdomains and task types.

In summary[1], our contributions are the following:

(1) We present the first of its kind interactive image generation technique combining GANs and relevance feedback for realistic image generation tasks.
(2) We demonstrate the technique as part of an interactive image generation tool and show that the technique is able to generate images that match user needs (as opposed to retrieving them) in response to relevance feedback obtained from the user.
(3) We empirically validate the approach in practical image generation and show performance improvements over conventional image retrieval in various image subdomains and task types yielding pragmatic performance.

## 2 RELATED WORK

A classic approach for finding images is based mainly on an image search paradigm: matching either the content of the images (search by content) or meta-data associated with the images (search by metadata) with an information need expressed by the user [28].

In this approach the corresponding image features or meta-data features are matched to an expression of information need. From the user's point of view, image search suffers from two fundamental challenges of matching the user's intention and semantics of the query to the images [14, 27]. That is, the expressions of information needs are difficult to construct and they may often not match the descriptors available for the images indexed in the system.

A well-explored technique for closing these gaps in interactive image search is relevance feedback [22]. In relevance feedback, the system solicits user feedback on the relevance of intermediate results to maximize relevant search outcomes over the course of an entire search session [23, 24]. At each step, the user selects images that partially match their needs, and the system returns images that better correspond to the images selected as partially relevant by the user in previous steps. In this way, the system can adjust the representation of the information needs to better match the internal representation of the image space. However, for the relevance feedback to be effective, the internal representation of the images must correspond to features that the user considers relevant. As conventional representation learning may not capture semantic features of the images, relevance feedback that relies on inconsistent feature representation may perform suboptimally.

Recent advances in deep feature extractors have allowed learning representations that have been shown to perform remarkably well in image classification [15]. They can also be used to derive textual descriptions of images by attending to features that humans use to detect objects and other image features [30]. These representations have also been shown to have high performance when applied in image retrieval applications [1].

However, even when equipped with a perfect representation of images, image search suffers from what we call retrieval limitation. Retrieval limitation means that the system is only able to provide the user with images that have been initially indexed in the system. Any information need that does not match what is already available in the indexed collection cannot be fulfilled. To tackle the retrieval limitation problem, researchers have shown an increasing interest toward generative models that are able to generate novel images that are not initially present in the indexed data. Particular success in generative models has been achieved with variational autoencoders [21] and GANs [8].

Recent works have shown impressive results in synthetic image generation [12]. However, many of the models have suffered from suboptimal control over the latent representation of the generated images, especially in what comes to matching the latent features with visual features. For example, it is nontrivial to learn a latent representation of human faces that would account separately for different visual characteristics of faces, such as eye color, hairstyle, or pose. StyleGAN architecture, which we also utilize, attempts to tackle this challenge by incorporating and building on progressive training to allow direct control of the strength of image features at different scales [13]. Such a model aims to ensure that the latent features representing the images would match stylistic features of the images and thus be closer to the semantic representation that humans might have about the images.

Research has combined generative models with interactive controls. Sketch-based image retrieval has been proposed, in which the user provides a hand-drawn sketch of the intended image output, and the generative model completes the image by using the shapes and other features present in the sketch [9, 17, 26]. Recent work has shown functional approaches to synthesizing photorealistic images from text [30, 33]. These take a textual description as input and generate an image with the features matching the textual input. However, these approaches have so far been successful only in rather limited domains, such as demonstrating the capability to generate plausible images of birds and flowers from textual descriptions. Generative models have also been used to translate features from one image to another [3, 7] and across text and images [10]. While very impressive, all previous approaches have a limited direct utility for information retrieval. This is because of two reasons. First, it is difficult for a user to control what the network produces, and any user control is implemented for a specific model in advance. Alternatively, the user has more control, such as in the text-to-image approach, but the models can only capture limited features in a single domain, such as photographs of human faces or flowers.

[1]Short video available at https://youtu.be/ZBYDKZRm-dE

## 3 INTERACTIVE GANs

In this section, we first give a formal treatment of the problem, then briefly describe technical aspects of GANs, and finally conclude by describing how to combine GANs and relevance feedback.

### 3.1 Theoretical formalization

A basic objective of an IR system is to retrieve documents that satisfy a user's information needs. Next, we formalize this process and use this formalization to contrast our approach with existing image retrieval systems.

Suppose there exists some latent high-dimensional space $Z \subseteq \mathbb{R}^n$ of all possible intents that a user might have. For a given search task, there exists at least one $\hat{z} \in Z$ that represents the specific intent of the task. Next, assume we have a finite collection $\mathcal{X} = \{x_1, \ldots, x_N\}$ of images. By viewing the images in $\mathcal{X}$ and interacting with an IR system, a user can provide feedback about the extent to which an $x \in \mathcal{X}$ matches the intent $\hat{z}$. We formalize this feedback as the function $d$, defined so that $d(x, \hat{z}) = 0$ when $x$ is by all standards a perfect match to $\hat{z}$. For images $x$ that are a less-suitable response to the intent $\hat{z}$, the function $d(x, \hat{z})$ takes a positive value that increases the more $x$ deviates from $\hat{z}$.

The objective of an IR system is thus to find the $x^* \in \mathcal{X}$ that is the best match to $\hat{z}$. More formally, we can say that an IR system aims to solve the following optimization problem:

$$x^* = \arg\min_{x \in \mathcal{X}} d(x, \hat{z}). \tag{1}$$

Because $\mathcal{X}$ is a finite set of images, this process can only find an $x^*$ that is the best alternative in $\mathcal{X}$. But even $x^*$ is unlikely to be a perfect match to $\hat{z}$, i.e., we almost always have $d(x^*, \hat{z}) > 0$. In other words, there are no perfectly matching images in $\mathcal{X}$.

What if we were able to produce a perfect image $x$ for which $d(x, \hat{z}) = 0$? Let $X \subset \mathbb{R}^m$ denote the space of all possible bitmap images (of some fixed size). Now we could think of solving the optimization problem in Eq. 1 over the set $X$ instead of the finite collection $\mathcal{X}$. But this is a difficult problem because the space $X$ is very sparse, as well as non-smooth. That is, only very few images in $X$ correspond to visually meaningful stimuli. The vast majority of images in $X$ show just "random" noise. Also, images that have similar content (e.g., two images of red sports cars) can be extremely far apart in $X$ with mostly noise images in the space in between.

Instead, the space $Z$ of all possible intents might have a structure that is better suited for optimization. First, there might not be any "noise" in $Z$ (i.e., every $z \in Z$ corresponds to some meaningful intent). Also, $Z$ might be smooth in the sense that whenever two intents $z$ and $z'$ are reasonably close to each other in $Z$ (in the Euclidean sense), they also have similar meanings. As $Z$ is for now only an abstract concept, we can assume that these properties hold by definition.

Denote a mapping from the latent space $Z$ of to all possible bitmap images by $G\colon Z \rightarrow X$. That is, given any intent $z \in Z$, the mapping $G$ returns the image $G(z) \in X$. Also, suppose that $G$ is *consistent* in the following sense: For every $z \in Z$, we have $d(G(z), z) = 0$. This means that for every possible intent $z$, the mapping $G$ returns the image $x \in X$, which is a perfect match to $z$ in terms of the function $d$.

Next, we define the subset of $X$ with images that each correspond to an intent in $Z$: We let $\mathcal{G}_Z = \{G(z) \mid \forall z \in Z\} \subset X$. Given this we can define an optimization problem that is a variant of Eq. 1:

$$\hat{x} = \arg\min_{x \in \mathcal{G}_Z} d(x, \hat{z}). \tag{2}$$

Because $G$ (by our assumption) satisfies the consistency property defined above, it must be the case that $\hat{x} = G(\hat{z})$ (i.e., $\hat{x}$ is a perfect match to the user's intent $\hat{z}$). The fundamental difference between the search problems in Eq. 1 and Eq. 2 is the set of images on which the IR system operates. In Eq. 1 this is the finite collection $\mathcal{X}$, while in Eq. 2 this is the (in theory) infinite set of images that correspond to the intents $z \in Z$.

**Problem definition:** The basic problem that we address in this paper is *how to efficiently find the solution $\hat{x}$ in Eq. 2.* Efficiently means that we want to find $\hat{x}$ without having to, for example, enumerate all images in the set $\mathcal{G}_Z$.

As a first step in our solution, we must find a practical way to implement the intent space $Z$ as well as the mapping $G$ from intents to images. These are discussed next.

### 3.2 Generative adversarial networks

We propose to implement the intent space $Z$ and the mapping $G$ with GANs [8]. A GAN is a machine learning technique for training models that can generate new objects from some *target domain*, most notably various kinds of images, as demonstrated already in [8] and subsequently by numerous authors, but also music [31], sounds [5], and even drug molecules [11]. Our focus is on synthetic photographs of faces, objects, or scenes. The training data of a GAN consists simply of a collection of examples from the desired target domain. Given a collection of portraits of people, for example, we can train a GAN to generate photorealistic portraits of people who do not exist [12, 13]. To an observer, such synthetic photographs can be indistinguishable from actual photographs.

**Technical background:** A GAN contains two components: a generator $G$ and a discriminator $D$. Formally, $G$ is a function that maps high-dimensional vectors from a *latent space* $Z \subset \mathbb{R}^n$ to an output space $X$. In our case points in $X$ are bitmap images. The discriminator $D$, on the other hand, is essentially a binary classifier for points in $X$. It aims to identify if a given image $x \in X$ belongs to the training data or if it has been generated by $G$.

Both $G$ and $D$ are trained simultaneously given a set of training documents. The training is set up as a kind of competition, where $G$ and $D$ aim to outsmart each other. The generator $G$ is given a set of $s$ random points $z_1, \ldots, z_s$ from the latent space $Z$. (Usually these are sampled from a standard multivariate normal distribution centered at the origin.) The outputs $G(z_i)$ are then fed to $D$ together with the real images from training data, and $D$ is trained to separate these two classes from each other. In practice both $G$ and $D$ are implemented as deep neural networks. This allows formulation of the learning problem in terms of a single differentiable loss function that can be minimized using efficient gradient-based techniques on graphics processing units (GPUs).

After $G$ has been trained, we can simply draw a random point $z$ from $Z$ and compute $G(z)$ to obtain a random image. We emphasize that $G(z)$ is *not* simply an image from the training corpus, it is a completely novel image. However, in practice, $G(z)$ will be an

image from the same "family" of images that the generator was trained on. That is, if the training data consisted of cat images, the generator could produce novel cat images, but not, for example, pictures of cars. Also, the generator cannot produce images that are wildly different from any of the training images even if they have some similar characteristics. For instance, a GAN trained on cat images could not generate an image of a cat playing a violin, unless such images had been part of the training corpus.

**Using $G$ and $Z$:** Returning to the optimization problem of Eq. 2, we propose to use the latent space of a GAN as the intent space $Z$ and the generator as the mapping $G$. Why would this be a good choice? In the discussion of Section 3.1 the intent space $Z$ is an abstract concept that is useful to define the problem we want to solve. However, for practical purposes, the mapping $G$, the GAN generator, is more relevant.

First, suppose that we use the latent space of a GAN as the intent space $Z$ and that there exists some vector $z \in Z$ so that the image $G(z)$ happens to be a perfect match to the user's intent $\hat{z}$. Then we can simply *define* that $\hat{z} = z$. Second, we argued that the intent space should exhibit some notions of non-sparsity as well as smoothness. These are, in the end, properties of $G$. If $G$ maps every point in $Z$ to some meaningful image, this implies that $Z$ is non-sparse. Likewise, if $G$ maps two neighboring points in $Z$ to images having mostly similar content, then $Z$ can be thought of as being smooth. The GANs used in our experiment have been trained with special techniques that aim to improve this aspect of their generators [13].

With the generator component of a GAN, we can thus create a concrete instance of the abstract problem of Eq. 2. Next, we discuss how this optimization problem can be solved using a simple relevance feedback approach.

### 3.3 Relevance feedback on GANs

Relevance feedback and the generator component $G$ of a GAN are easily combined to provide a practical solution to Eq. 2. In short, the idea is to let the user navigate in the latent space $Z$ of the generator $G$ until they reach the point $\hat{z}$ so that the image $G(\hat{z})$ matches their intent.

**Basic setup:** As an example, suppose that the generator $G$ has been trained to generate photographs of different types of cars in different orientations and contexts. We assume that our user needs a photograph of *a modern red compact car that sits in a parking lot so that the image shows both the front and driver's side of the car* and that this intent is captured by the vector $\hat{z} \in Z$, as explained above. In what follows we call $G(\hat{z})$ the *target image*.

How do we find the latent $\hat{z}$? The dimensions of the latent space $Z$ of a GAN cannot, in general, be associated with any real-world features of the resulting photographs, such as color, shape, or orientation of the car. Nor are there any other means for us to directly pinpoint where in $Z$ there are vectors that the generator $G$ would map to images having some particular characteristics, let alone the specific intent of our user. We solve this problem with an interactive navigation system, where navigation is controlled by a simple relevance feedback mechanism.

**Relevance feedback steps:** The user starts from the location $q_1 \in Z$, where $q_1$ can be chosen either randomly or by letting the user select an image from a collection generated by $G$ from random points in $Z$. Let $q_i$ denote the position of the user in $Z$ at the start of the $i$:th feedback step. On every step $i$, the system first presents the user with $k$ images $G(z_1), \ldots, G(z_k)$. These are generated from points $z_1, \ldots z_k$ in $Z$ that are chosen randomly from the vicinity of $q_i$ (details below). From these, the user selects as feedback those that have some matching features with the target image. The feedback images are then used to update the position $q_i$ with the well-known Rocchio algorithm (see e.g. [19] for details). That is, we calculate the average of the vectors of the feedback images, denote the resulting vector by $z_{\text{avg}}$, and update the vector $q_i$ by letting $q_{i+1} = (1 - \alpha)q_i + \alpha z_{\text{avg}}$. The location $q_i$ should now move step-by-step in $Z$ and eventually approach the point $\hat{z}$, which yields the desired target image $G(\hat{z})$. Here, $\alpha$ controls the rate at which the vector $q$ moves. Smaller values mean shorter steps.

To see a real example of this process from the system we implemented, Figure 2 shows a sequence of candidate images together with the selected feedback images (indicated by the green bars) from a randomly picked task in our experiment. We can see how the candidate images progressively become more similar to the desired target image of the task in question (the red car shown in the top part of Fig. 1).

**Generating candidate images:** In practice, we sample $z_1, \ldots, z_k$ from a multivariate normal distribution centered at $q$, with a diagonal covariance matrix $\Sigma = I$ (here $I$ denotes the identity matrix). The dispersion parameter $s$ determines how far from $q$ the candidate images are. If set to a value too large, the candidate images will become too diverse, and likewise, if $s$ is too small, the candidate images will all be very similar to each other. For the experiment, we found a value for $s$ that produced good candidates by manual exploration.

## 4 EXPERIMENT

To test the methodology described in Section 3, we conducted a user experiment in which participants generated images in response to pre-defined tasks. The experiment followed a within-subject design. All participants were exposed to 16 image generation tasks that were presented in randomized order.

### 4.1 Participants

We recruited 34 volunteers to take part in the study by posting recruitment advertisements online. Complete data were obtained for 29 participants, of which 19 were male and 10 were female. Their ages ranged from 21 to 53 years with a mean age of 30.9 years. The participants' educational background was 7% doctoral degrees, 45% master's degrees, 38% bachelor's degrees and 10% other. The participants were provided with an informed consent, ensuring that before continuing to the actual experiment the participants understood how the data were to be used and were aware of their rights as participants. The participants received information that the experiment was designed in accordance with the Declaration of Helsinki and that the participants had the right to withdraw at any time without negative consequences. The participants were not trained in person to use the system, but they were provided with text instructions accompanied with an example video explaining the functionality of the system.

| Model | Target description |
|---|---|
| *bed* | Black bed with two pillows and red sheets with stripes, no window, light comes from a lamp |
| *cat* | White cat with stripes and blue eyes lying down |
| *car* | Large old green car on asphalt on a sunny day |
| *face* | Asian man, age 60, frowning while looking left, wears eyeglasses |

**Table 1: Textual targets used with the different models.**

## 4.2 Models, task types, and task definitions

The tasks used in the experiment were parametrized by the model (*cat*, *car*, *bed*, *face*), type of target definition (*image* and *text*) and type of task (*near* and *far*). This resulted in $4 \times 2 \times 2 = 16$ different tasks in total, and every participant completed each of these. The basic objective for all tasks was the same: *use relevance feedback to generate an image that matches the given target description as closely as possible.*

**Models:** We used four different pre-trained image generators made publicly available[2] by the authors of [13]. These four models capture four different classes of photographs: people, animals, objects, and scenes. In the following we refer to the models by the names *cat*, *car*, *bed*, and *face*, and they generate synthetic images of cats, cars, bedrooms, and human faces, respectively. The GANs had been trained with the following data: Flickr-Faces-HQ dataset at 1024×1024, LSUN Bedroom dataset at 256x256, LSUN Car dataset at 512×384, and LSUN Cat dataset at 256x256. All models used a 512-dimensional latent space from which vectors were mapped to bitmap images. The generated images had the same size as images in the training data for the respective model.

**Task definition:** We considered two types of task definitions: *target images* and *textual descriptions*. In case of target images, the participants were simply shown an image generated by the model in question. The full set of target images used in the experiment is shown in the left columns of Figure 5 (see caption for details). Textual descriptions were constructed as an alternative way of specifying a target. These are shown in Table 1. The description specified a number of target image features that users were instructed to ensure are in the resulting image. Note that the textual and visual target descriptions are different from each other.

**Task type:** To investigate if participants can use the system both to refine a reasonably good image and generate the desired image from a starting point that shares no common features with the target, we considered two task types: *near* and *far*. A *near* task was a task in which the initial image already matched the task, and the participants were instructed to further improve the image. In this case the starting image was chosen by the participants from a grid of 100 candidate images generated by the respective models, of which a few were pre-selected to be fairly good matches to the intent of the task. In contrast, in a *far* task the initial image shared almost no or little visual similarity with the target. These starting images had been manually selected by the authors. See Section 4.4 for more details on how baselines were chosen for both task types.

[2] https://github.com/NVlabs/stylegan

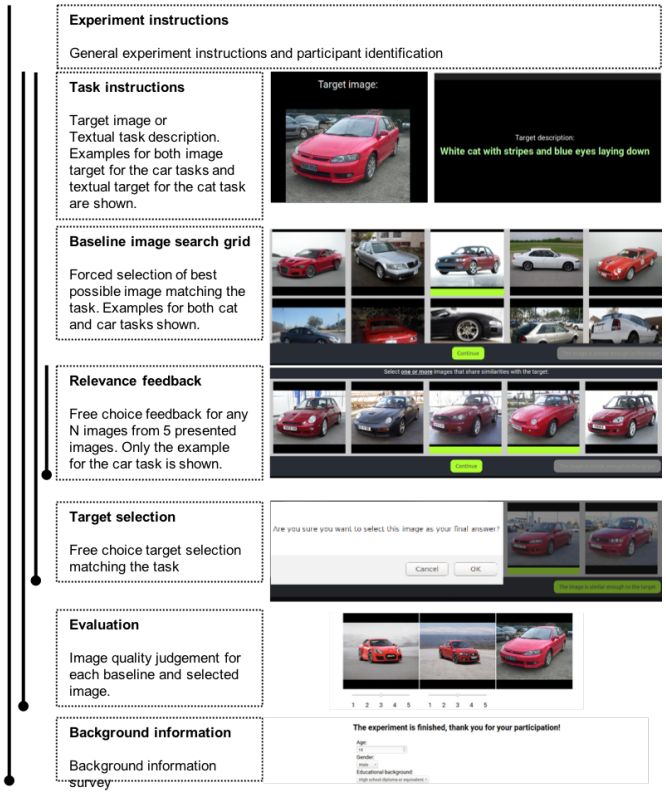


**Figure 1: The complete experiment flow illustrated with examples. The black lines on the left indicate phases. The nested phases are repeated within the experiment.**

## 4.3 Procedure

The basic protocol for each experiment scenario was the following. After signing up for the experiment, the participants were sent an email with instructions on how to participate in the experiment. The instructions contained an overall explanation on the purpose of the experiment and links to the online experiment and an instructional video.

The instructional video illustrated an example task along with the instructions on how to use the system, provide feedback, and assess information returned by the system. The video lasted less than 5 minutes and demonstrated an example task that was not part of the experiment.

Then, the participants entered the actual experiment, which consisted of completing 16 image generation tasks that were presented in randomized order. The overall flow of the experiment is shown in Figure 1. Each *near* task started with a description of the target accompanied by an image search result grid that was the same for all participants. The participants were asked to select the best matching image from the grid. This part of the protocol was designed to mimic a basic image retrieval system. For *far* tasks, this step was skipped, as there was only a single starting point.

Next, the participants were provided with an interactive relevance feedback control, shown in more detail in panel A of Figure 2. The participants were able to select one or several images as

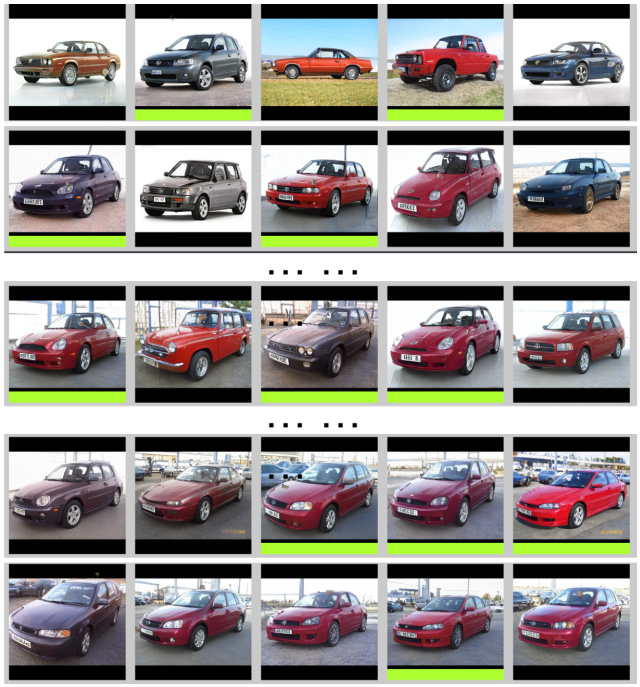

**Figure 2: An example of a relevance feedback sequence for a randomly picked car task. The first two, a middle, and the last two steps are shown. At every step, the user selects the closest matching images underlined with a green bar. The picture of the red car selected as the answer is underlined with a green panel in the bottom row. The task instruction is the same as the target image in Figure 1. Note that all shown images are generated by a GAN.**

relevance feedback. The target image or text was always shown at the top of the user interface. The set of candidates shown on every step was computed from the relevance feedback given in the previous step, as described in Section 3.3. For example, the list of candidates shown in the second row of Fig. 2 was generated by our system based on the images selected in the first row (marked by a green bar).

This process was repeated until the participant was satisfied with the results and indicated it by pressing a designated button or terminated if the participant used more than 40 feedback iterations. During the experiment, the participants had full freedom to use the provided system as they wished, and they were not forced to provide feedback or continue using the system longer than they wanted.

Third, after completing all tasks, the participants entered an evaluation phase (see again the bottom part of Figure 1). In this phase, the participants were presented with the image they started from and the generated image they chose after the relevance feedback interaction. The participants were then asked to rate on a scale of 1 (worst) to 5 (best) the quality of the images based on how satisfied they were with the image given the task description. These ratings are used to evaluate the results in Section 5.

Finally, each participant filled in a post-task questionnaire to collect background information and ensure that the experiment worked technically and that the participant understood the tasks. The participants received a movie voucher worth approximately $15 as a compensation for their time. The completion of the experiment took on average 38 minutes.

### 4.4 Baseline images

**Baselines for *near* tasks:** In the experiment one of the goals was to compare the results of our approach to a *baseline image search approach*. For this purpose, each model was used to generate a set of 100 images corresponding to a typical image search output. Distorted and unrealistic images were filtered out and replaced with new generated images until all of the images looked realistic enough to be results from an image search engine. Images were then ensured to be partially relevant to the tasks. That is, for both task definitions (*image* and *text*) this set of baseline images contained a number of partial matches.

For example, for the *cat* model, the textual task definition was "white cat with stripes and blue eyes laying down", and it was ensured that the set of baseline images contained images that satisfied some of the desired features. In particular, at least one baseline image matched the given intent so that *all but one* feature was present. However, none of the baseline images matched all possible features to ensure that the participants were not satisfied with the initial image and could not exit the task without using the relevance feedback to improve the image.

**Baselines for *far* tasks:** For the *far* tasks, there was only a single baseline image from which the users started their navigation. These were chosen to have as few (or no) similarities as possible with the task intent. For example, for the textual task definition "white cat with stripes and blue eyes lying down", the preselected starting image showed a black cat without stripes, with yellow eyes, and standing up.

### 4.5 Apparatus

The experiment was run on a virtual machine instance in the Google Cloud Platform. The instance was powered by a single Nvidia Tesla P100 GPU and an Intel Xeon 2.3GHz CPU. It was powerful enough to generate five candidate images in at most one second for each feedback step, making interactions with the system smooth. The participants were instructed to connect to the instance through its public IP address using their own laptop or desktop device.

## 5 RESULTS

The results are shown in Figures 3 and 4, illustrated with a full example for a random participant in Figure 5. Detailed results with statistical analysis are shown in Table 2, where the baseline (BL) and GAN columns show the average rating for the baseline and GAN-generated images, respectively. The $\Delta = GAN - BL$ column shows absolute improvement in quality, while $p$ shows the Bonferroni-corrected p-value of a two-sample t-test between all BL and GAN scores. We find that in all cases, the difference $\Delta$ is positive, and with the exceptions of *bedroom* images (both task definitions, *near* task) and *face* images (*text* definition, *near* task), this difference is significant at a level of 0.001.

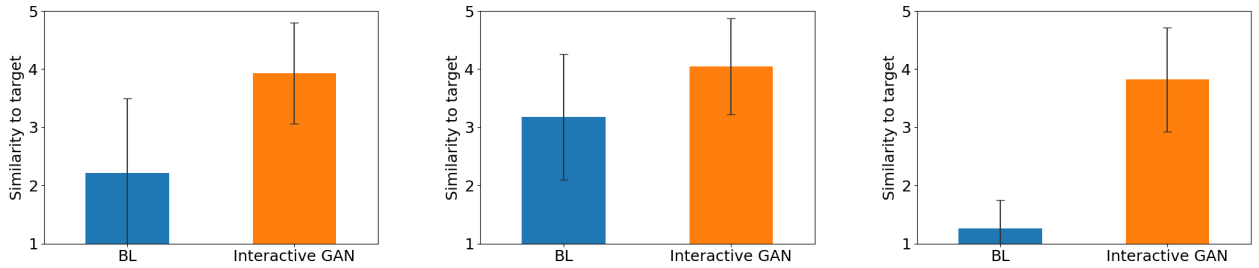


**Figure 3: Main results. The left panel shows the average rating (as evaluated by the participants) aggregated over all task types for baseline and generated images. The middle and right panels show the same for the near and far task types, respectively.**

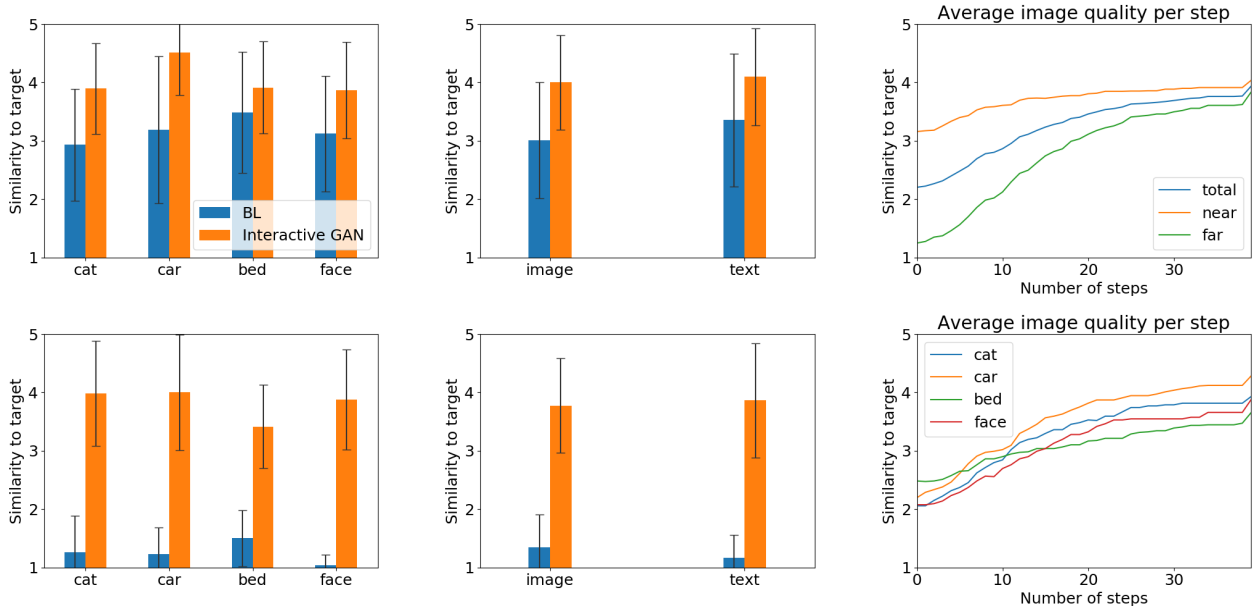


**Figure 4: Results by model (left column), task definition (middle column), and progressively by steps (right column). In the left and middle column, the upper row shows the results for the *near* task and the lower row for the *far* task. In the right column, the upper row shows the results split by task type (*far* or *near*), and the lower row shows the results split by model.**

Below, we discuss first the main findings and then results separately for different models (cat, car, bed, face), task definitions (image, text), and temporal effects with respect to relevance feedback iterations.

## 5.1 Main findings

Our main findings are shown in Figure 3. When considering the average of all task types (leftmost panel in Fig. 3), we find that there is a significant difference in the image ratings when comparing baseline images with those generated by the interactive GAN approach. This difference is more pronounced for the *far* task type (rightmost panel in Fig. 3) than for the *near* type (middle panel in Fig. 3). Finally, no substantial differences were found on the ratings of GAN-generated images between the *near* and *far* tasks.

There are two important observations to be made here. First, in the *near* tasks, despite the starting point being already a good match with the target description, the participants were able to still improve the images by a significant margin. Second, in the *far* tasks, even if the starting image bore very little or no resemblance to the target description, the participants were able to generate images that were as good as those they found in the *near* task.

## 5.2 Model effects

We first analyzed whether the feedback performed consistently with the different models. Model effects are shown in Figure 4 (left column). The upper figure shows the results for the *near* task type and the lower for the *far* task type. Significant differences were found for image and text targets for all models except for the bedroom model. For the other models, the results were consistent, with

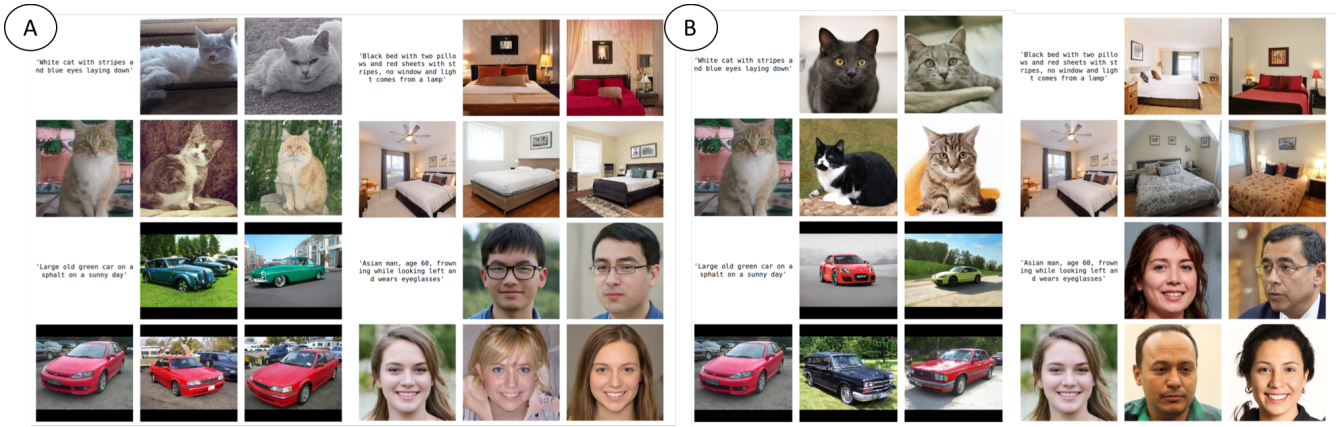


**Figure 5: Example results from a randomly sampled user for all tasks. The *near* condition results are shown in panel A and the *far* condition search tasks are shown in panel B. Both panels A and B contain four tasks. For each task, the left column shows the task description–either the *image* or *textual* target. The middle column shows the starting image, and the right column shows the resulting image after the GAN relevance feedback.**

the best performance obtained for the face model. The other models significantly improved over the baseline in all task types and both task definition conditions. The results suggest that the model did not significantly affect the performance, except for the bedroom model, and the image generation process seems to generalize over different models. However, non-significant differences in performance for the bedroom model suggests that StyleGAN, which was used in the experiments, may suffer from problems for specific types of input (scenes).

## 5.3 Task definition effects

We further analyzed whether the way the task was presented to the participants had an effect on the results. Task definition effects are shown in Figure 4 (middle column). The results were consistent in both task definition conditions and in line with the main results. The *far* task showed significantly better performance than the *near* task, and no difference was found between the task definition types. The results suggest that the performance was independent from the way the task was presented to the participants, and image generation output was significantly better than the baseline images in both types of task definitions.

## 5.4 Temporal effects

The previous results establish that the participants were able to find images that they consider to be a good match with the given target description independently from the model or the way the task was presented to the user. However, these results do not indicate how long it takes on average to find a good image. We analyzed the temporal effects separately for the different models and task types. Figure 4 (right column) shows the average rating as a function of the number of feedback steps and separately for the *far* and *near* task types and different models. For the *near* tasks, most of the improvement occurs in the first 10 steps, while for *far* tasks, up to 25 steps of feedback are required to reach the target image. This suggests that while participants successfully generate high-quality images in both task types, there is a significant increase in user effort when the tasks are more difficult to complete.

## 5.5 An illustrative example

A full example for a random participant for all tasks is shown in Figure 5. The results for the *near* task are shown in panel A, and the results for the *far* task are shown in panel B. All resulting images match the task descriptions, and the improvements are visually apparent in both tasks and most salient in the case of the *far* task. The quality of the images in the case of the *bedroom* model is clearly weakest as also indicated in the quantitative data.

# 6 DISCUSSION

Next, we discuss both our empirical findings and their implications.

**Empirical findings:** Our findings strongly suggest that explicit relevance feedback on GANs is a viable method for generating images that satisfy a particular information need. Participants of our experiment were able to both further improve an already matching image and generate the desired image from an unrelated starting point. They were able to do this with both visual and textual targets, and without any prior training. Moreover, the results were not substantially affected by the types of images.

Results for the *bedroom* tasks were clearly the weakest. This may have been caused by the bedroom model being less well trained than the others. As discussed in sections 3.1 and 3.2, the latent space should ideally satisfy a "smoothness" property so that there are no abrupt transitions in the visual features of images generated from two neighboring vectors $z$ and $z'$. Perhaps the latent space induced by the *bedroom* model was not as smooth as the others. Conversely, our results suggest that the latent spaces induced by the *car*, *face*, and *cat* models were relatively smooth, as otherwise the relevance feedback method would have not allowed successful navigation.

It is possible that, despite our best efforts, the dispersion parameter of the distribution from which candidate images were sampled was poorly set for the *bedroom* model. This is partly supported by

**Table 2: The overall results for all models, task types, and task definitions. The results are shown for the baseline with standard deviation, GAN with standard deviation, improvement in image quality between the baseline and GAN, and Bonferroni corrected p-value and the respective F-statistic.**

| | Image definition | | | | | | | Text definition | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Near | | | | | | | Near | | | | | | |
| | BL | $\sigma_{BL}$ | GAN | $\sigma_R$ | $\Delta$ | p-value | F | BL | $\sigma_{BL}$ | GAN | $\sigma_R$ | $\Delta$ | p-value | F |
| nmCats | 2.90 | 0.98 | 4.00 | 0.60 | 1.10 | $p < 0.001$ | 26.9 | 2.97 | 1.12 | 3.79 | 0.94 | 0.83 | $p < 0.001$ | 9.3 |
| nmBeds | 3.45 | 0.99 | 3.83 | 0.76 | 0.38 | $p = 1.0$ | 2.7 | 3.52 | 0.95 | 4.00 | 0.80 | 0.48 | $p = 0.65$ | 4.4 |
| nmCars | 2.83 | 1.07 | 4.38 | 0.90 | 1.55 | $p < 0.001$ | 35.6 | 3.55 | 1.35 | 4.66 | 0.48 | 1.10 | $p < 0.001$ | 17.1 |
| nmFaces | 2.86 | 0.88 | 3.79 | 0.86 | 0.93 | $p < 0.001$ | 16.7 | 3.38 | 1.05 | 3.93 | 0.80 | 0.55 | $p = 0.44$ | 5.1 |
| *Mean* | 3.01 | 0.98 | 4.00 | 0.78 | 0.99 | $p = 0.027$ | 20.5 | 3.35 | 1.12 | 4.09 | 0.76 | 0.74 | $p = 0.018$ | 9.0 |

| | Image definition | | | | | | | Text definition | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Far | | | | | | | Far | | | | | | |
| | BL | $\sigma_{BL}$ | GAN | $\sigma_R$ | $\Delta$ | p-value | F | BL | $\sigma_{BL}$ | GAN | $\sigma_R$ | $\Delta$ | p-value | F |
| nmCats | 1.34 | 0.55 | 3.79 | 0.73 | 2.45 | $p < 0.001$ | 209 | 1.17 | 0.38 | 4.17 | 0.66 | 3.00 | $p < 0.001$ | 449 |
| nmBeds | 1.79 | 0.68 | 3.72 | 0.59 | 1.93 | $p < 0.001$ | 134 | 1.21 | 0.41 | 3.10 | 1.05 | 1.90 | $p < 0.001$ | 82 |
| nmCars | 1.17 | 0.38 | 3.79 | 1.05 | 2.62 | $p < 0.001$ | 160 | 1.28 | 0.53 | 4.21 | 0.90 | 2.93 | $p < 0.001$ | 228 |
| nmFaces | 1.07 | 0.26 | 3.79 | 0.86 | 2.72 | $p < 0.001$ | 266 | 1.00 | 0.00 | 3.97 | 0.87 | 2.97 | $p < 0.001$ | 341 |
| *Mean* | 1.34 | 0.47 | 3.78 | 0.81 | 2.43 | $p < 0.001$ | 192 | 1.16 | 0.33 | 3.86 | 0.87 | 2.70 | $p < 0.001$ | 275 |

the lower right panel of Figure 4, which shows that the bedroom images improve at a slower rate than the other models. This is an indicator of the relevance feedback algorithm struggling to make progress through the latent space. Finally, giving useful relevance feedback for the interior scenes may also simply have been more difficult for the participants than with portraits or photos of animals or objects.

The issues discussed above in the case of the bedroom model could, of course, be encountered when applying the same approach with new models. A useful area for future investigations would thus be to train GANs in a manner that would make their latent spaces even more easy to navigate.

**Implications:** To our knowledge, this is the first study that investigates the use of relevance feedback for controlling a GAN to interactively *generate* instead of retrieving relevant images. Our approach enables generation of images that can meet user needs even in cases when the underlying image collection does not contain a sufficient matching image.

The models we experimented with are, to some extent, "toy examples". To serve as a practically useful tool to augment (or even replace) image search on the web, the GAN would have to be trained with hugely diverse and large training data. Such a model is probably beyond state-of-the-art adversarial learning for the near future. However, assuming that such a GAN could eventually be trained, the approach we propose would enable users to generate arbitrary images. While current image editing software already allow manipulation of images in complex ways, these approaches require specific tools and expertise and are not direct alternatives to our approach.

Despite the advantages, technology for generating synthetic images of human faces, objects, or scenes may also have potentially harmful applications. Generating images provokes problems of fake or unreal content to be generated and exploited in precarious ways. One defending aspect of GANs and other generative models has been that it is not easy to have control over their output, and generating images that are intentionally harmful has been difficult or required significant investment of human labor.

However, our work demonstrates that it is surprisingly straightforward to combine GANs with relevance feedback and utilize such models in an interactive loop between the user and the generative model. This approach turns out to be effective in the generation of images that match user intentions by allowing relatively precise control over what the generator produces. While this opens ingenious opportunities for supporting creative human processes, it may also permit unethical use of the technology.

## 7 CONCLUSIONS

We set out to study if it is feasible to generate images rather than search them from a fixed collection. We did this with a methodology similar to existing image search systems that use explicit relevance feedback. In practice, we combined a relevance feedback algorithm with image generators based on GANs. We chose a very simple relevance feedback algorithm to keep the design of the system as straightforward as possible. Our experiment shows that a) users were able to successfully generate relevant images with our system, and b) the resulting synthetic images can be more relevant to the user's information needs than a baseline image chosen from a fixed collection.

## ACKNOWLEDGMENTS

The research was supported by the Academy of Finland (Decisions 313610,322653,328875,1314262,1308946). We thank the University of Helsinki IT for Science staff for their excellent assistance with GPUs.

# REFERENCES


[1] Artem Babenko, Anton Slesarev, Alexandr Chigorin, and Victor Lempitsky. 2014. Neural Codes for Image Retrieval. In *Computer Vision – ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (Eds.). Springer International Publishing, Cham, 584–599.

[2] Yang Cao, Hai Wang, Changhu Wang, Zhiwei Li, Liqing Zhang, and Lei Zhang. 2010. MindFinder: Interactive Sketch-based Image Search on Millions of Images. In *Proceedings of the 18th ACM International Conference on Multimedia (MM '10)*. ACM, New York, NY, USA, 1605–1608. https://doi.org/10.1145/1873951.1874299

[3] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. 2018. StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[4] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z. Wang. 2008. Image Retrieval: Ideas, Influences, and Trends of the New Age. *ACM Comput. Surv.* 40, 2, Article 5 (May 2008), 60 pages. https://doi.org/10.1145/1348246.1348248

[5] Chris Donahue, Julian J. McAuley, and Miller S. Puckette. 2018. Synthesizing Audio with GANs. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*. https://openreview.net/forum?id=r1RwYIJPM

[6] Hao Fang, Saurabh Gupta, Forrest Iandola, Rupesh K. Srivastava, Li Deng, Piotr Dollar, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John C. Platt, C. Lawrence Zitnick, and Geoffrey Zweig. 2015. From Captions to Visual Concepts and Back. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[7] Abel Gonzalez-Garcia, Joost van de Weijer, and Yoshua Bengio. 2018. Image-to-image translation for cross-domain disentanglement. In *Advances in Neural Information Processing Systems 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.). Curran Associates, Inc., 1287–1298. http://papers.nips.cc/paper/7404-image-to-image-translation-for-cross-domain-disentanglement.pdf

[8] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. 2014. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*. 2672–2680.

[9] Longteng Guo, Jing Liu, Yuhang Wang, Zhonghua Luo, Wei Wen, and Hanqing Lu. 2017. Sketch-Based Image Retrieval Using Generative Adversarial Networks. In *Proceedings of the 25th ACM International Conference on Multimedia (MM '17)*. Association for Computing Machinery, New York, NY, USA, 1267–1268. https://doi.org/10.1145/3123266.3127939

[10] L. He, X. Xu, H. Lu, Y. Yang, F. Shen, and H. T. Shen. 2017. Unsupervised cross-modal retrieval through adversarial learning. In *2017 IEEE International Conference on Multimedia and Expo (ICME)*. 1153–1158. https://doi.org/10.1109/ICME.2017.8019549

[11] Artur Kadurin, Alexander Aliper, Andrey Kazennov, Polina Mamoshina, Quentin Vanhaelen, Kuzma Khrabrov, and Alex Zhavoronkov. 2017. The cornucopia of meaningful leads: Applying deep adversarial autoencoders for new molecule development in oncology. *Oncotarget* 8, 7 (2017), 10883.

[12] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. 2018. Progressive Growing of GANs for Improved Quality, Stability, and Variation. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. https://openreview.net/forum?id=Hk99zCeAb

[13] Tero Karras, Samuli Laine, and Timo Aila. 2019. A Style-Based Generator Architecture for Generative Adversarial Networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. 4401–4410.

[14] Christoph Kofler, Martha Larson, and Alan Hanjalic. 2016. User Intent in Multimedia Search: A Survey of the State of the Art and Future Challenges. *ACM Comput. Surv.* 49, 2, Article 36 (Aug. 2016), 37 pages. https://doi.org/10.1145/2954930

[15] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (Eds.). Curran Associates, Inc., 1097–1105. http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[16] Wenhao Lu, Jingdong Wang, Xian-Sheng Hua, Shengjin Wang, and Shipeng Li. 2011. Contextual Image Search. In *Proceedings of the 19th ACM International Conference on Multimedia (MM '11)*. ACM, New York, NY, USA, 513–522. https://doi.org/10.1145/2072298.2072365

[17] Yongyi Lu, Shangzhe Wu, Yu-Wing Tai, and Chi-Keung Tang. 2018. Image Generation from Sketch Constraint Using Contextual GAN. In *The European Conference on Computer Vision (ECCV)*.

[18] Mathias Lux, Christoph Kofler, and Oge Marques. 2010. A Classification Scheme for User Intentions in Image Search. In *CHI '10 Extended Abstracts on Human Factors in Computing Systems (CHI EA '10)*. ACM, New York, NY, USA, 3913–3918. https://doi.org/10.1145/1753846.1754078

[19] Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. Introduction to information retrieval. Cambridge university press.

[20] Neil O'Hare, Paloma de Juan, Rossano Schifanella, Yunlong He, Dawei Yin, and Yi Chang. 2016. Leveraging User Interaction Signals for Web Image Search. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '16)*. ACM, New York, NY, USA, 559–568. https://doi.org/10.1145/2911451.2911532

[21] Yunchen Pu, Zhe Gan, Ricardo Henao, Xin Yuan, Chunyuan Li, Andrew Stevens, and Lawrence Carin. 2016. Variational Autoencoder for Deep Learning of Images, Labels and Captions. In *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (Eds.). Curran Associates, Inc., 2352–2360. http://papers.nips.cc/paper/6528-variational-autoencoder-for-deep-learning-of-images-labels-and-captions.pdf

[22] Yong Rui, Thomas S. Huang, Michael Ortega-Binderberger, and Sharad Mehrotra. 1998. Relevance feedback: a power tool for interactive content-based image retrieval. *IEEE Trans. Circuits Syst. Video Techn.* 8 (1998), 644–655.

[23] Tuukka Ruotsalo, Giulio Jacucci, and Samuel Kaski. 2019. Interactive faceted query suggestion for exploratory search: Whole-session effectiveness and interaction engagement. *Journal of the Association for Information Science and Technology* (2019).

[24] Tuukka Ruotsalo, Giulio Jacucci, Petri Myllymäki, and Samuel Kaski. 2014. Interactive intent modeling: Information discovery beyond search. *Commun. ACM* 58, 1 (2014), 86–92.

[25] Tuukka Ruotsalo, Jaakko Peltonen, Manuel J. A. Eugster, Dorota Głowacka, Patrik Floréen, Petri Myllymäki, Giulio Jacucci, and Samuel Kaski. 2018. Interactive Intent Modeling for Exploratory Search. *ACM Trans. Inf. Syst.* 36, 4, Article Article 44 (Oct. 2018), 46 pages. https://doi.org/10.1145/3231593

[26] Patsorn Sangkloy, Jingwan Lu, Chen Fang, Fisher Yu, and James Hays. 2017. Scribbler: Controlling Deep Image Synthesis With Sketch and Color. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[27] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. 2000. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22, 12 (Dec 2000), 1349–1380. https://doi.org/10.1109/34.895972

[28] Bart Thomee and Michael S. Lew. 2012. Interactive search in image retrieval: a survey. *International Journal of Multimedia Information Retrieval* 1, 2 (01 Jul 2012), 71–86. https://doi.org/10.1007/s13735-012-0014-4

[29] Tung Vuong, Miamaria Saastamoinen, Giulio Jacucci, and Tuukka Ruotsalo. 2019. Understanding user behavior in naturalistic information search tasks. *Journal of the Association for Information Science and Technology* 70, 11 (2019), 1248–1261.

[30] Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. 2015. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37 (ICML'15)*. JMLR.org, 2048–2057. http://dl.acm.org/citation.cfm?id=3045118.3045336

[31] Li-Chia Yang, Szu-Yu Chou, and Yi-Hsuan Yang. 2017. MidiNet: A Convolutional Generative Adversarial Network for Symbolic-Domain Music Generation. In *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017, Suzhou, China, October 23-27, 2017*. 324–331. https://ismir2017.smcnus.org/wp-content/uploads/2017/10/226_Paper.pdf

[32] Zheng-Jun Zha, Linjun Yang, Tao Mei, Meng Wang, Zengfu Wang, Tat-Seng Chua, and Xian-Sheng Hua. 2010. Visual Query Suggestion: Towards Capturing User Intent in Internet Image Search. *ACM Trans. Multimedia Comput. Commun. Appl.* 6, 3, Article 13 (Aug. 2010), 19 pages. https://doi.org/10.1145/1823746.1823747

[33] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N. Metaxas. 2017. StackGAN: Text to Photo-Realistic Image Synthesis With Stacked Generative Adversarial Networks. In *The IEEE International Conference on Computer Vision (ICCV)*.

[34] Xiang Sean Zhou and Thomas S. Huang. 2003. Relevance feedback in image retrieval: A comprehensive review. *Multimedia Systems* 8, 6 (01 Apr 2003), 536–544. https://doi.org/10.1007/s00530-002-0070-3