[Draft]Interaction Design for Systems that Integrate Image Generation Models

Victor Dibia
victor.dibia@gmail.com
Researcher
Santa Clara, California, USA

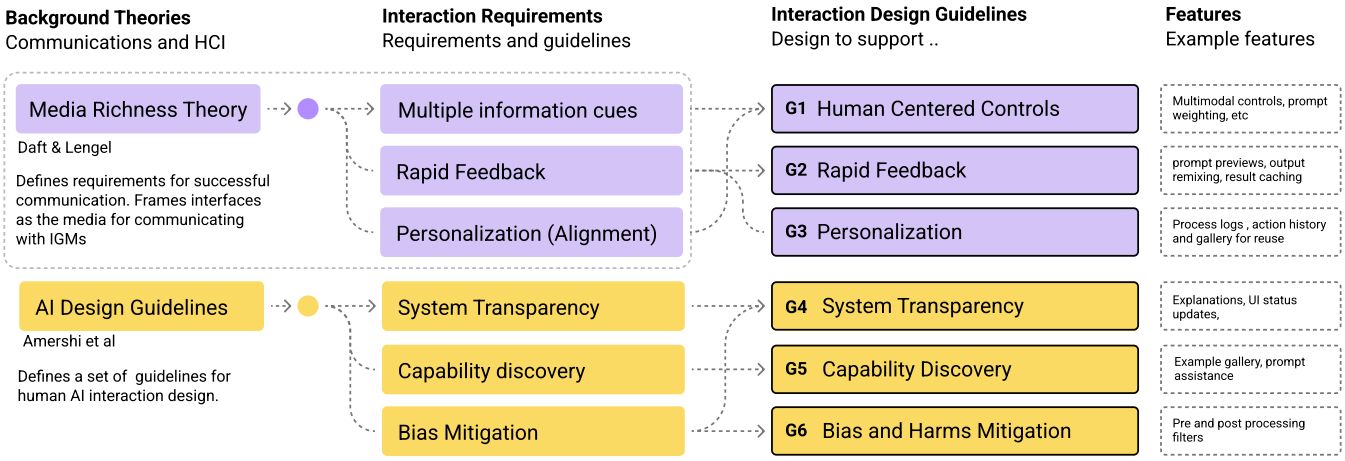

Figure 1: In this work we draw on communication theory (media richness theory [2]) and research on Human-AI interaction design [1] in defining a set of interaction guidelines for systems that integrate image generation models (IGMs).

ABSTRACT

Recent advanced multimodal image generation models (**IGM**'s) have enabled the creation of high quality images on consumer compute devices, with implications for new creative applications. As developers build new experiences that integrate these models, it is still unclear how *best to align* the vast technical capabilities of these models with the goals and values for users. This work addresses this *alignment/control* problem in two ways. First, we draw on theories from communications (media richness theory) and human-ai interactions in deriving a set of high level guidelines for building **IGM**'s - designing for *human centered controls*, *rapid feedback*, *personalization*, *transparency*, *capability discovery* and *harms mitigation*. Next, we present *Peacasso* - an open source user interface that implements these guidelines, with features including a generic image generation editor (text/image prompting, inpainting, outpainting), process and outcome curation, sense making tools (prompt preview and explanations) and a set of high level application presets (tiling, video generation, story generation). *Peacasso* provides a flexible python api useful for experimentation and is available on Github.

Conference acronym 'XX, June 03–05, 2018, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

KEYWORDS

datasets, neural networks, gaze detection, text tagging

ACM Reference Format:
Victor Dibia. 2018. [Draft]Interaction Design for Systems that Integrate Image Generation Models . In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX)*. ACM, New York, NY, USA, 7 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

The ability to convert natural language text to high quality images that capture concepts expressed in the text has been the focus of multiple computer vision research streams over the last decade. Systems that integrate these capabilities have potential to improve the *image creation* user experience by converting a difficult task (e.g., illustrating images is high cognitive load and high effort) into a simple task (e.g., describing an image in natural language is low cognitive load, low effort). Recent advances in text-to-image generation models, specifically diffusion models such as DALLE-2 [8], Stable Diffusion [9], Midjourney [11] have shown particularly compelling results - they generate high quality images, at high resolution, on consumer compute devices purely from text. In essence, these advances inspired the proliferation of new tools and systems where an AI model works in concert with a human in co-creating visual

imagery. There have now been plugins that integrate **IGM**'s into existing design tools like Adobe Photoshop [12] as well as brand new tools such as Dream Studio [1] etc.

Within these applications, a human provides some representation of the goal of the collaboration (typically via some input mechanism such as text or other modalities, see section 2.2), and the AI model produces visual imagery *guided* by this input. Achieving these goals typically involves the creation of some *system* - an interface that mediates communication between the *human*, and the **IGM***model* - towards achieve a set of goals for the human. Note that the *interface* may implement additional integrations with a set of *external entities* as needed. A systems perspective is valuable, important given that the interface component plays a profound role in shaping the socio-technical outcomes of HumanAI collaboration *beyond* the core technical capabilities of the underlying model. For example, pairing an **IGM**with frame curation and text curation tools and genreation pipleline can lead to the generation of videos or illustrated books; pairing an interface with appropriate capabilty discovery mechanisms can reduce time to productivity or quality of output.

In practice, developing systems that integrate machine learning models and enable *CoCreation*faces a few challenges. Specifically the probablistic nature of machine learning models make them unpredictable when used as design materials, which can result in disruptive, confusing or harmful experiences [1]. Furthermore, while the underlying models are highly capable (i.e., can generate complex, novel imagery), a significant effort is still required of the user to craft the right prompt that steers the model towards a desired output. In this work, we argue that these challenges can be address to some extent by improving the interaction interface through which humans interact with the underlying models.

Overall the contributions of this work are summarized as follows

- A set of 6 high level design guidelines for building **IGM** interfaces - designing for *human centered controls*, *rapid feedback*, *personalization*, *transparency*, *capability discovery* and *harms mitigation.*. These guidelines are informed by existing theory - we adapt insights frommedia richness theory that highlights requirements for effective communication and guidelines for human-ai interaction [1].
- We contribute Peacasso - with features including a generic image generation editor (text/image prompting, inpainting, outpainting), process and outcome curation, sense making tools (prompt preview and explanations) and a set of high level application presets (tiling, video generation, story generation). While *Peacasso* provides a UI interface (see Figure 3), it also provides a flexible backend api designed to enable experimentation on novel interfaces for **IGM**'s.It is built in the open and benefits from feedback and iterations that occur as the core models and community practice of **IGM**'s evolve.

2 RELATED WORK

Systems that integrate **IGM**s consist of multiple components - image generation models, a core user interface and third party dependencies. In this section, we image generation models and mental models for users.

[1]Dream Studio https://beta.dreamstudio.ai/dream

2.1 Image Generation Models

The ability to generate high quality novel images has been the target of extensive research in the deep learning and computer vision field. Generative adversarial networks (GANs) showed us that we could implicitly learn the distribution of data and draw samples from this distribution; VAEs showed us how to explicitly map data to a (gaussian) distribution and learn the parameters of this distribution. However, we could only generate small blurry images - 64x64px; and when we got better at larger images, we still had diversity and quality problems and at very high computational cost. While these models enabled free form generation of novel images, the inference setup offered little control (random noise vector used as a latent code used to sample from the learned distribution in generating an image). To advance on this, researchers explored the use of conditional generation, where discrete labels corresponding to dimensions were associated with the training process and used during inference to control generations along these dimensions Engel et al. [3], Radford et al. [7].

Recently, there has been the introduction of diffusion models - models that learn a noising process (adding noise to some input across several steps until that input is indistinguishable from pure noise) and a denoising process that reverses the noising process (removes noise across several steps until a legit image is recovered). The name diffusion comes from a branch of physics (non-equilibrium thermodynamics) which describes how particles may move from areas of high concentration to lower concentration in order to achieve equilibrium. In practice, while diffusion models yield very high image quality, many challenges (slow sampling due to a large number of denoising steps etc) have had to be addressed to make this process usable in image generation - with landmark papers like , DDPM [4], DDIM [10] etc.

More recently latent diffusion models [9] were introduced and address an important challenge with diffusion models - their memory requirements. They achieve this by proposing we apply the noising and denoising process in latent space (instead of raw image/pixel space) and then map this latent space back to the image space. This dramatically reduces the memory requirements for running diffusion models, hence the competitive results we have seen recently on consumer GPUs. Finally, these models address the issue of control by leveraging advances from large multimodal (text+image models e.g. CLIP Radford et al. [7]) where generation is conditioned on free form text representation from these multimodal models.

2.2 Control Modalities for Image Generation Models

To facilitate the generation of arbitrary images, **IGM**'s must learn an efficient latent representation that covers the dataset on which they are trained. The resulting $latent representation$s is treated as a distribution that can be sampled from using a $noise vector$or $latent code$. In practice, the degree to which this learned representation (aka latent space) can be navigated or ***controlled*** is directly correlated with their usefulness [6].

- **Latent Code**. This line of research explores methods for manipulating some *existing* latent code that results in predictable outcomes. For example, learning operations on $latent code$for a generated image that might interpolate features associated with

Table 1: Input modalities provided by image generation models can be categorized in terms of the control precision that they provide to the user and the ease with which humans can author or interprete those modalities (human intelligibility).

Input Modality	Sub Category	Description	Precision	Human Intelligibility
Random seed	Random seed	Seed used to generate latent code	low	low
Latent Code	Random seed	Latent code used to sample latent space	low	low
Text	Text category	Class name for class conditioned generation	medium	high
	Natural language	Natural language descriptions used to guide image generation model	medium	high
	Prompt natural language	Natural language specific to prompting syntax	high	medium
	Weighted prompt natural language	Applying weighted combinations of text descriptions	high	high
Image	Image	Initializing latent code from image	high	high
	Text + Image	Natural language guidance + image control	high	high
	Text + Image + mask	Text + image constrained to a region in the input image	high	high

images in human datasets such as like age, gender [5] or dataset invariant attributes such as scale and translation Plumerault et al. [6].

- **Text Input**. This approach involves guiding the generation image generation using representations learned from text. For example, diffusion models may start with some arbitrary random noise vector based on a *seed*, and iteratively refine the generated image in a manner that is conditioned on text represenation from a multimodal modal. While this approach provides user agency to describe a scene in natural language sentences, there is limited control Text is by definition underspecied, it the information contained in 5 or 50 words must be converted into an image .. a scene that can be represented in 1,000 words. Furthermore, while these models do support natural language, effective prompts that generate high quality images have been shown to be *less natural*, often requiring the use of specialized modifier keywords, art styles and descriptors that vary with model and training dataset.
- **Image Input** : In this regime, images are provides as input to the model. In the case of diffusion models, $latentcode$is derived based on the input image and used as the starting point for the diffusion process. In addition, a image + mask approach can be applied where parts of this input may be masked and the model task to inpaint or complete only a region of the input image.

These control modalities can also be classified in terms of human intelligibility (the ease with which a human an author control artifacts) and control precision (the level of control that an mechanism offers over the generated image). For example, a simple random latent code lets the user generate a random image, the only guarantee being that the image conforms to properties of the dataset from which the model was trained. On the other hand natural language allows the user to describe salient concepts that are in turn represented in the generated image. On the other hand image input paired with text description provides control on concepts in the image and opportunity to improve specific sections of the image (in painting).

In this work, we explore how input modalitied offered by **IGM**s map to design decisions and resulting guidelines.

2.3 Community Practices on Image Generation Systems

While advances in

3 INTERACTION DESIGN: REQUIREMENTS AND GUIDELINES

Existing studies have explored requirements for good interaction across multiple parties. In this section, we focus on two of these studies. First we explore interaction requirements for human-human communication [2]. Second, we explore interaction requirements for human-AI interaction [1]. We then use these insights to derive a set of high level design guidelines for systems that integrate **IGM** models.

3.1 Requirements for Effective Communication: A Media Richness Theory Perspective

in 1986, Daft and Lengel [2] began studying drivers for *effective communication* media within organizations. Their work, known today as media richness theory describe rich media as one that fosters understanding, learning and minimizes ambiguity or uncertainty [2]. They argue that media richness is a function of several characteristics including - ability to handle multiple information cues simultaneously - ability to facilitate rapid feedback, ability to establish a personal focus, ability to utilize natural language, minimize uncertainty (examples, capability discovery). In turn, rich media drives *effective communication* by reducing uncertainty especially when processing equivocal information (information that may be subject to multiple interpretations). This observation is especially important in the context of **IGM** systems where the output of the model is subject to multiple interpretations e.g., a prompt can result in ***multiple valid images***. While *media richness theory* aims to explain information processing behaviors between humans (or organizations), it can be applied in framing communication challenges that arise when humans work with **IGM** systems. Specifically, we frame the **IGM** interface as a communication medium and argue that richness of the interface directly impacts the effectiveness of the collaboration between human and AI. By revisiting the characteristics of rich media, we can derive a set of interaction requirements for **IGM** systems.

3.1.1 Ability to Handle Multiple Information Cues Simultaneously. Daft and Lengel [2] give cite face-to-face communication as rich media given the they offer a party the ability to integrate multiple cues such as body language, tone of voice, personal contact, and convey messages in *natural language*. Similarly, an **IGM** interface should strive towards providing multiple rich modalities that are human-intelligible and easy to author such as natural language text,

images, audio and video(see table 1 on intelligbility of model input affordances).

Guideline 1: Human Centered Control

(1) Support multiple input modalities that are *human-intelligible* and *easy to author* such as natural language text, images, audio and video.
(2) Support the combination of multiple modalities towards improving human agency and fine grained control of results

3.1.2 Ability to Facilitate Rapid Feedback. Daft and Lengel [2] argue that rich media facilitate rapid feedback by allowing for immediate feedback and response; in turn, this allows users to converge on a common interpretation. Similarly, an **IGM** should offer affordances for rapid, timely feedback and iteration on ideas. This is especially important for design with **IGM**'s where each generation may take some time and often requires costly GPU compute. For example, a user should be able to compare results, select results of interest, make edits, request new results or interupt the model as needed.

Guideline 2: Rapid Feedback

(1) Support a *iterative creative* process composed of *actions* where the output of each *actions* may be used as input for other actions.
(2) Support methods that help the user make sense of how their input choices may impact the output of the model with minimal resources (time and compute).

3.1.3 Ability to Establish a Personal Focus. Finally, Daft and Lengel [2] find that communications that were personalized and focused on the individual were more effective. Similarly, an **IGM** interface should support fine grained control in representing their preferences and intent. For example, providing methods specifying a variety of concepts or styles to be represented in the result can enhance level of control av'ailable to the user. Also, a good **IGM** interface should support users as they switch between various mental models.

Users can have one of two distinct high level mental models and associated goals as they interact **IGM**systems. This categorization of user behaviour is not novel and has been mirrored on multiple creativity or search tools [].

Understanding these mental models can help in understanding the underlying user goals and informing the appropriate set of interaction affordances (workflows) that can be offered to users.

- **Exploratory Mental Model** In the exploratory mindset, the user comes to to tool with a limited understanding of the exact tasks. First, they
  limited information and expertise - they may not know the exact outcome they wish to accomplish, may be limited in artistic skill, limited in the capabilities of the model. Despite all of this, their goal is to engage in the creative process ..extract high quality imagery that allows th particularly have a clear idea of their goals,
- **Exploitative Mental Model**
  In this model, users have a clear idea of what they are interested in creating. The bulk of their effort is geared towards representing their intent using the ineterface provided by the system such that their goal is achieved.

Guideline 3: Personalization

(1) Support multiple mental models based on the user's needs (e.g.,)exploration and exploitation .
(2) Support low level exploration as well as high level application workflow presets.

3.2 Guidelines for Human AI Interaction

While an MRT lens allows us to identify communication requirements for an **IGM**system, one limitation is that it assume human-human communication, hence a need to account for human-ai communication requirements. To address this, we turn to insights from research on design guidelines for human ai interaction by Amershi et al. [1]. Amershi et al. [1] acknowledge that results from AI models are performed under uncertainty leading to unpredictable outcomes that may be disruptive, offensive or unsafe. provide a set of interaction guidelines that underpin scenarios where humans interact with AI models. In this section we revisit a subset of those guidelines contextualized to the image generation scenario.

To enable cocreation, the user must understand the core capabilities of tan AI model and be appropriately trained on how to elicit those capabilities. For example, it is usefulf for the user to understand that the model's behaviour is restricted to the knowledge within the trainig dataset. Interaction breaks down when the user demands when the user is not aware of the limiations of the model and attempts to requests tasks beyond the model's ability or consistently under utilizes the model.

Guideline 4: Transparency

(1) Provide tools that help the user make sense of the model's capabilities and limitations.

Guideline 5: Capability Discovery

(1) Support controls that help the user discover and understand the capabilities of the model.

Guideline 6: Bias and Harm Mitigation

(1) Support controls that minimize the risk of results that reinforce bias or harm.

4 PEACASSO

In this section we discuss the design and technical implementation of *Peacasso* - a tool and user interface to support experimentation with **IGM**s. *Peacasso*implements concrete features informed by the design guidelines discussed in section 3 (see Fig 1). It consists of the following components: - a set of ***core modules***, a ***web api*** and a front end ***user interface*** as shown in Figure 2 .

4.1 Core Modules

Core modules focus on implementing interfaces for querying the underlying stabile diffusion model. This includes a generic pipelines class that enables the user to generate images across multiple modes

4.1.1 (Unified) Image Generator Pipeline. This module implements a single unified image generation pipeline that is parameterized and exposes all model functionality - text to image, image to image, mask+image to image ie. inpainting. This design approach is memory friendly as only a single version of each model component is loaded and reused as needed. Input signature to this module is defined by a $GeneratorConfig$ data model which is shared across the backend api and the front end interface as the primary signature for generating images.

4.1.2 Applications Helper. This module implements several application workflows that may call the unified image generation pipeline on demand (or process its outputs) in addressing specific application workflows. For example, a tiling application workflow may take a $PromptConfiguration$ as input or an image, recompose a call to the generation pipeline .

4.1.3 Prompt Helper. This module implements a set of functions to support the user in crafting prompts used for generation. It currently includes a prompt predictor model trained on example stable diffusion prompts.

4.1.4 Backend Web API. This includes backend and frontend sections. The backend web api is built with Fastapi and provides rest api endpoints for interacting with the core modules (requesting generations). Two server end points are provided: (a) a rest based service endpoint that calls the unified generator end point. This endpoint is decoupled from other services to allow the service run on a dedicated server that can be provisioned, scaled and managed independently. (b) A web socket based service endpoint that provides a real time interface for interacting with the service. This endpoint is used by the front end to provide a real time interface.

4.2 UI Interface Modules

The user interface is built with React (ant design and tailwind css) and provides a UI interface for interacting with the backend.

4.2.1 Generator Controls. A set of controls to configure the generation pipeline and request images from the backend. All of the controls are parameterized by the $GeneratorConfig$ data model.

4.2.2 Prompt Editor. Provides a set of tools for finegrained control over the prompt used for generation. This includes standard interfaces for editing text, image and mask prompts.

- **Remixing**: The Editor supports **remixing** where each generated image can be reused as a prompt.
- **Prompt Mixing**: Sliders for weighted prompt combination where weights are assigned to multiple prompts
- **Prompt explanation**: Provides a list of words or a word tree related to each token in the provided prompt. This capability serves to illustrate how the CLIPText model assigns context to the words in the prompt (e.g. does it see the word glass in terms of a material or in terms of it being a lens.)
- **Prompt suggestion**: Suggests completions to prompts based on predictions from the prompt helper core module.
- **Prompt interpolation**: Suggests completions to prompts based on predictions from the prompt helper core module.
- **Prompt previews**: For a given prompt, provide a preview of images that are likely to be generated by those prompts. This feature is implemented by providing a nearest neighbour search on generated preexisting prompts and images (possibly using a service like lexica.art). This can provide rapid feedback in exploring multiple prompts before incure costs associated with new generations.

4.2.3 Session Gallery. This feature enables the user to start a session which records a series of **actions**. Actions may be generation requests or application workflow requests. A session must be explicitly started and may be automatically stopped based on some time constraints. This provides a shareable log for both their process and outcomes for a given session with benefits in reproducibility of results.

4.2.4 Applications. Implements presets for a set of applications that may involve image generation. This includes presets for generating tiled images, automated outpainting, video generation, latent space exploration and more.

5 FUTURE WORK

5.1 Opportunities in Improving Control

While this work has looked at interface design features that allow the user control the generated output of the model, these approaches may ultimately be limited by the model's capabilities or complexity of UI steps needed to exert control. To address this, there may be other avenues to explore such as:

5.1.1 Prompt Sensitive Training. Perhaps some approach to structuring training data where a model must learn to function at best effort where text was small and text was large. Also, traning paradigms that improve sentitivity to salient words may improve cotnrol. For example, crafting datasets for descriptions with high syntax overlap but low conceptual overlap are equally representated can enalbe the model learn the improtance of salient tokens and reflect ths in generation.

5.1.2 Domain/Style Specific Models. While the current state of the art models are trained on a wide variety of domains, there may be value in training models on a specific domain or style. By constraining a model to a specific style, the burden of control and space for uncertainty is reduced. For example, a model trained on a specific style of art may be able to generate images that are more consistent with the style.

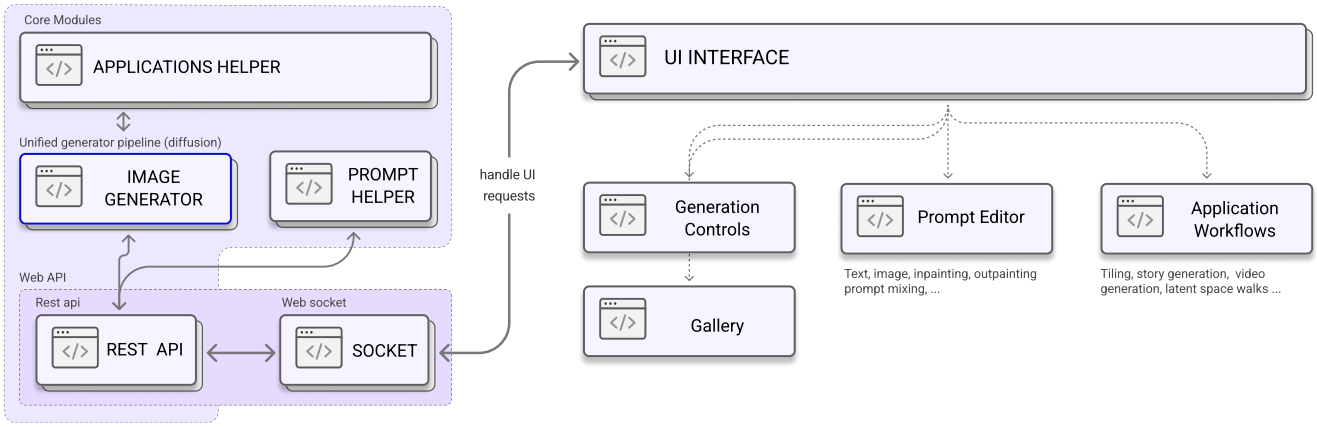

Figure 2: Architecture diagram for the *Peacasso* library.

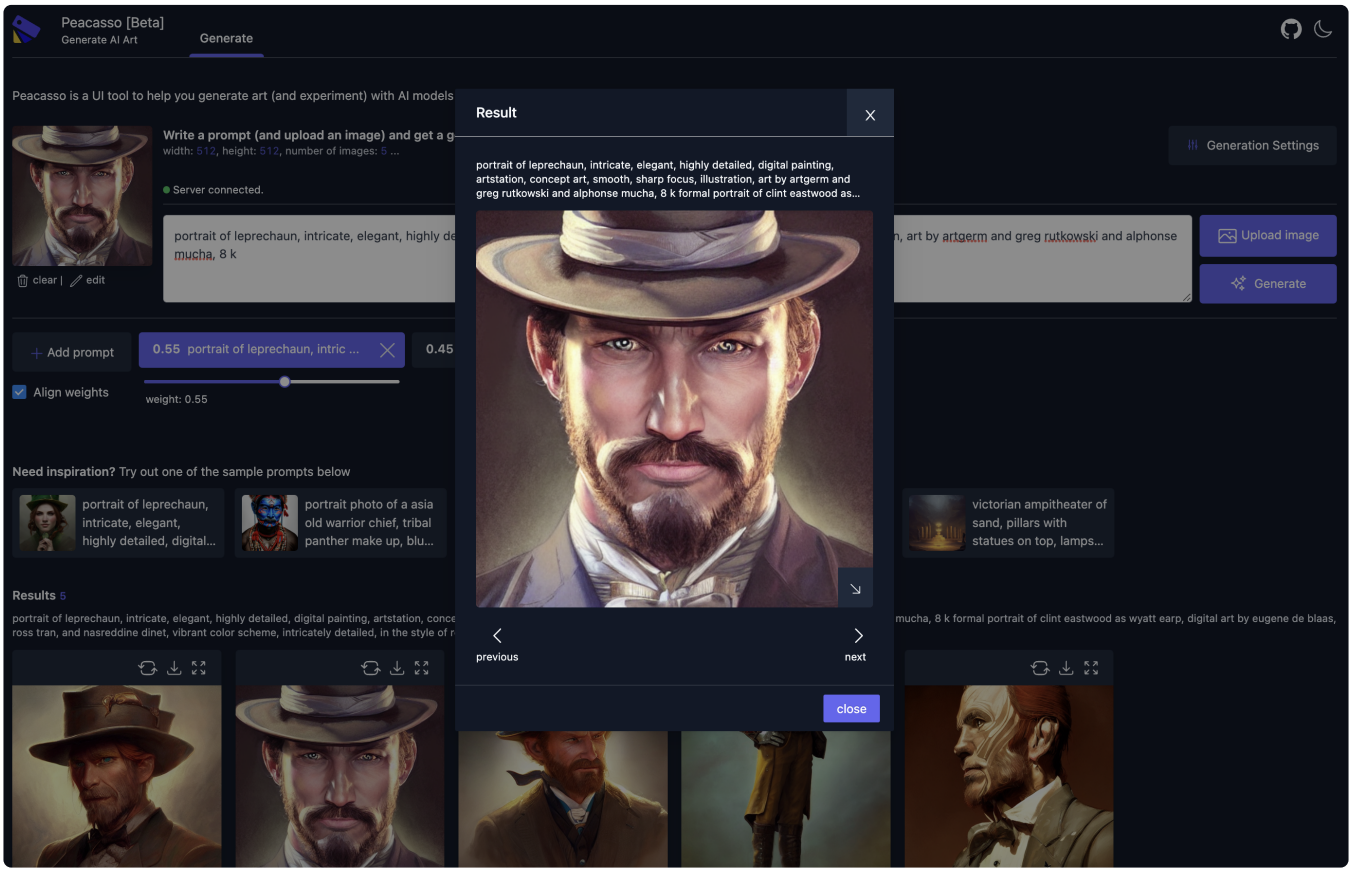

Figure 3: Screenshot of the Peacasso Interface. The user can "remix" a generated image as input to a new generation request.

5.2 Image Quality Evaluation

Do authors have the ability to evaluate the quality of the images they generate? Hisotrically, artistic style are a particular skill/craft that is honed over time, leaving the human in a position where they assess a test for which they are neither skilled nor versed.

6 CONCLUSION

In this work, we leveraged insights from established research in communication theory [2] and human ai interaction design [1] in deriving a set of high level design guidelines for systems that integrate image generation models. To provide a concrete realization of these guidelines, we contribute *Peacasso*- an open source library

that implements a user interface for image generation workflows and provides a flexible python api suitable for experimentation. We hope these contributions will achieve two main goals: i) inform the creation of human-centered user interfaces for the wide variety of **IGM** systems being created today ii) provide a prototyping tool (*Peacasso*) that supports research and industry applications of image generation models.

ACKNOWLEDGMENTS

Thanks to Gonzalo Ramos, Rick Barraza and Sharon Lo for valuable conversations and feedback on the capabilities of image generation models.

REFERENCES

[1] Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul N Bennett, Kori Inkpen, et al. 2019. Guidelines for human-AI interaction. In *Proceedings of the 2019 chi conference on human factors in computing systems*. 1–13.

[2] Richard L Daft and Robert H Lengel. 1986. Organizational information requirements, media richness and structural design. *Management science* 32, 5 (1986), 554–571.

[3] Jesse Engel, Matthew Hoffman, and Adam Roberts. 2017. Latent constraints: Learning to generate conditionally from unconditional generative models. *arXiv preprint arXiv:1711.05772* (2017).

[4] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems* 33 (2020), 6840–6851.

[5] Tero Karras, Samuli Laine, and Timo Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4401–4410.

[6] Antoine Plumerault, Hervé Le Borgne, and Céline Hudelot. 2020. Controlling generative models with continuous factors of variations. *arXiv preprint arXiv:2001.10238* (2020).

[7] 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. In *International Conference on Machine Learning*. PMLR, 8748–8763.

[8] 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* (2022).

[9] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10684–10695.

[10] Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502* (2020).

[11] Midjourney team. 2022. Midjourney Community Showcase. https://www.midjourney.com/showcase/.

[12] Stability team. 2022. The Stability Photoshop plugin. https://exchange.adobe.com/apps/cc/114117da/stable-diffusion.