Ideation via Critic-Based Exploration of Generator Latent Space

Puneet Jain, Najma Mathema, Jonathan Skaggs and Dan Ventura
Computer Science Department
Brigham Young University
Provo, UT 84602 USA
puneetj@byu.edu, nmathema@byu.edu, jbskaggs12@gmail.com, ventura@cs.byu.edu

Abstract

We present a system for generating, evaluating, and refining logos that can act as a collaborator for creating relevant logo designs. The system combines computational vision and language systems to generate logo design ideations that suitably represent company identity as expressed in a tagline or with keywords. The generative part of the system employs a Generative Adversarial Network (GAN) architecture, while the evaluative part makes use of two vision-based classification models and a language embedding model to assess how well generated logos align with the identity of the company. This process is iterated by using feedback from the evaluation module to guide exploration of the latent space of the GAN. The results may be used as is, or further curated/refined by human designers. For evaluation of the results, two surveys with different sets of participants are conducted. Findings show the utility of feedback-mediated latent space search and that participants rate the system-generated logos above average on creativity and relevance.

Introduction

Generative artificial intelligence (AI) has become quite popular and recently developed models have produced impressive results in domains including visual, musical and linguistic artifacts. Current state-of-the-art approaches for such generative tasks include generative adversarial networks (GANs) (Goodfellow et al. 2014), variational autoencoders (VAEs)(Kingma and Welling 2014) and transformers (Vaswani et al. 2017). All of these approaches leverage deep learning architectures that are trained to minimize loss over a large dataset of artifact examples and generate new artifacts by, effectively, interpolating between abstractions of that training data. Such approaches have been shown to produce very good results, in the sense that the generated artifact is a *typical* representative of the artifact class (i.e, a person's face, a song, a story); however, none of these approaches account for the *value* of the output, in the sense typically considered by computational creativity (CC) researchers [i.e., is it an interesting face?, or a sad song? or an entertaining story? cf. Ritchie (2007)]. This is not surprising, given the generative AI agenda, but the question naturally arises whether such models might be somehow incorporated into CC systems. This paper explores this question by proposing the idea of iterative generation guided by a critic, where both the generator and the critic are deep-learning models that interact via a semantic vector space. To make the ideas concrete, we use them to implement a system for logo creation; we want a system that generates typical logos but also one that generates valuable logos—logos that represent a target company well.

A company logo is something like a visual "meme", something that incorporates, represents and communicates the company's identity concisely, visually, and cleverly. The design of a good logo is a time-consuming task requiring not only graphic design skills but also creative thinking and possibly even a bit of serendipity. We present a computational system for logo generation that, while it can be used as a completely autonomous system, may perhaps most appropriately serve as a creative collaborator for human designers, and we give a proof-of-concept of using it in that capacity here—the system iteratively brainstorms and refines a pool of logos for several (fictitious) companies from which we make a final selection.

We are not aware of any extant systems that tackle the task of intentionally creating a logo that serves as a visual representation of identity and/or that communicates concepts. However, there are some systems that tackle similar or related tasks, some of which help inspire our approach here.

For example, Özbal, Pighin, and Strapparava developed a system called BrainSUP to support brainstorming creative sentence generation. It was intended to be used collaboratively or as a support tool by a human creator and offers that user/collaborator the ability to constrain the search space in various ways to ensure the sentence communicates desired concepts (2013).

Using a similar framework to BrainSUP, Tomašič, Papa, and Žnidaršič used evolutionary computation to build a more autonomous system for creating company slogans. They used eight different evaluation functions to guide the generation, with an aim to produce a slogan reflecting, in some way, the company identity (2015).

On the visual front, Heath, Norton, and Ventura created a computational artist called DARCI which produces images with the intention of communicating one or more linguistic concepts (2014).

More recently, Cunha, Martins, and Machado have explored blending emojis to communicate pairs of linguistic

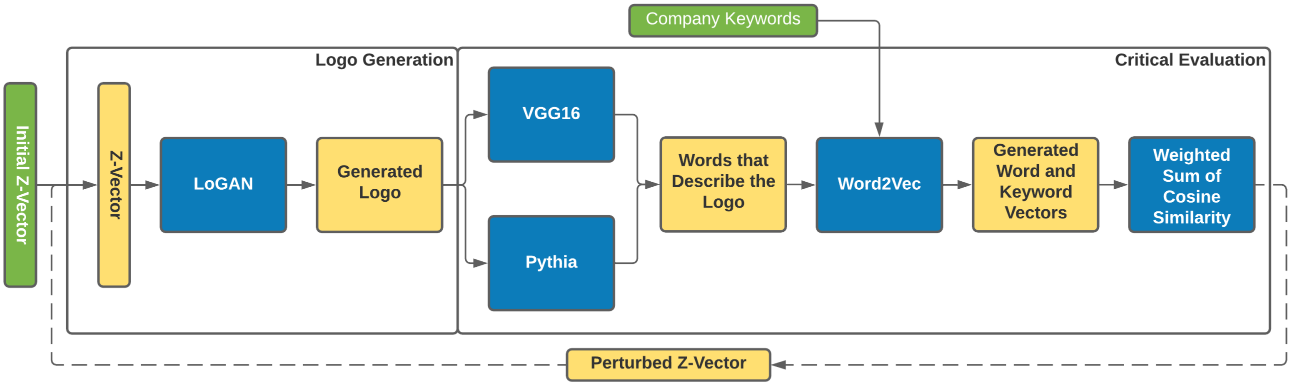


Figure 1: Model architecture for automatic logo generation. The system is initialized with random z-vectors, generates a logo for each using a GAN, evaluates the resulting images based on target keywords, and uses the z-vectors associated with the best images as feedback to further explore the GAN's latent space.

concepts visually (2018). The same research group have also recently introduced a system for creating visual blends of a country's flag with another symbol connected to current events relevant to that country (Cunha et al. 2020).

As for logo generation itself, Li, Zhang, and Li apply shape grammars for generating logo designs automatically. The authors encode design knowledge using rule sets and validate the feasibility of their approach via an experiment (2017).

Atarsaikhan, Iwana, and Uchida make use of neural style transfer with clip art and text to generate logos given an input image, introducing a new distance-based loss function to preserve the silhouettes of text and objects (2018).

Sage et al. tackle the problem of logo generation using GANs. Because GANs are data hungry, they scraped the web to create a logo dataset named Large Logo Dataset (LLD) with 600K+ images. They then used this dataset to train a GAN for generating plausible logos, employing synthetic labels obtained through clustering to help stabilize the GAN and prevent mode collapse (2018). Mino and Spanakis built on the work of Sage et al. to condition the network based on color so that a user of this system could have more creative control, ultimately developing a GAN conditioned on 12 different colors (2018).

We build on this recent work, making use of the Mino and Spanakis model as a blackbox generator trained on the Large Logo Dataset (Sage et al. 2018) and drawing a little from the ideas of Atarsaikhan, Iwana, and Uchida, to develop an aesthetic evaluation metric so that the system can autonomously determine the relevance of a generated logo. The result is a system that takes as input the name of the organization and a few keywords that describe the company objective and outputs a set of 100 relevant logos which could be used to inspire ideas (or used as is) for the company's visual identity. Contributions of this work include the following:

1. demonstration of iterative guided exploration of the GAN latent space for exploitation/convergent design thinking
2. demonstration of the use of GANs for exploration/divergent design thinking
3. demonstration of computational creativity (CC) system-building using modern, off-the-shelf models
4. autonomous incorporation of linguistic concepts into the visual design
5. collaborative/autonomous creation of intentional visual identity in the form of a logo

Model Architecture

The process of designing a logo is expensive and labor intensive. The purpose of our work is to build a creative system that can automate some of the logo generation process, providing suggestions from which a customer might make a final decision or perhaps acting as a collaborator with a human designer.

A diagram of the overall system architecture is shown in Figure 1. At a high-level, the system is composed of two modules: the *generator* and the *critic*. The generator takes as input a z-vector which acts as a seed for the logo generation process and is passed as input to the LoGAN model (Mino and Spanakis 2018), which produces a candidate image. The critic takes as input a candidate image and a set of keywords that describe the target identity for the customer company. It then uses two vision-based classifiers—VGG16 (Simonyan and Zisserman 2015) and Pythia (Singh et al. 2018)—that each provide a classification label for the image and a corresponding confidence in that classification. It uses the word embedding model Word2Vec (Mikolov et al. 2013) to vectorize the classification labels and the company keywords and then computes cosine similarity between the keywords and labels to assign a score for the candidate image. Based on this score, the original input z-vector is perturbed, and the process repeats. Pseudocode for the two modules is given in Algorithms 1 and 2, and further details are discussed below.

Algorithm 1 CREATELOGO(K)

```
1:  Input: keywords $K$
2:  $z^* \leftarrow$ randomnormal$(0, 1)$
3:  $x^* \leftarrow$ LoGAN$(z^*)$
4:  $\nu^* \leftarrow$ EvaluateLogo$(x^*)$
5:  while not done do
6:      $z \leftarrow z^* +$ randomnormal$(0, 0.3)$
7:      $x \leftarrow$ LoGAN$(z)$
8:      $\nu \leftarrow$ EvaluateLogo$(x, K)$
9:      if $\nu > \nu*$ then
10:         $x^* \leftarrow x$
11:         $z^* \leftarrow z$
12:         $\nu^* \leftarrow \nu$
13: return $x^*$
```

Algorithm 2 EVALUATELOGO(x, K)

```
1: Input: image $x$, keywords $K$
2: $\omega_p, \gamma_p \leftarrow$ Pythia$(x)$
3: $\omega_v, \gamma_v \leftarrow$ VGG16$(x)$
4: $\delta_p \leftarrow$ cosinesim(word2vec$(\omega_p)$, word2vec$(K)$)
5: $\delta_v \leftarrow$ cosinesim(word2vec$(\omega_v)$, word2vec$(K)$)
6: $\nu \leftarrow \frac{1}{Z}(\gamma_p \delta_p + \gamma_v \delta_v)$
7: return $\nu$
```

Logo Generation

LoGAN is a Generative Adversarial Network that has learned to generate logos (Mino and Spanakis 2018). It takes as input a 128-dimensional z-vector and returns a generated logo. A z-vector is a point in an (arbitrary, condensed) feature space. The power of GANs is their ability to learn to organize this latent space such that it is well-behaved, in the sense that each point maps to a reasonable generated artifact and proximal points in z-space map to generated artifacts that are similar (in their original/natural feature space). In the LoGAN model, the latent z-space is therefore an abstraction of the set of possible logos. Each unique z-vector will generate a different logo, and similar z-vectors will generate similar logos. We explore this latent space by initially choosing a random z-vector and then iteratively refining it to perform a local search in its neighborhood. For efficiency and to facilitate diverse exploration, we operate the model on a batch of pools of z-vectors, somewhat reminiscent of speciation in evolutionary computation. We initially randomly generate 100 pools of 100 z-vectors (line 2 of Algorithm 1), each of which is used as input to the GAN, resulting in the generation of 100 pools of 100 candidate images (line 3). Each of these images is then passed to the critic for evaluation (line 4).

Critic Evaluation

After a batch of pools of logos is generated, the logos are passed through two image recognition systems (VGG16 and Pythia), which classify the images (one label each, ω_v and ω_p, respectively) as well as producing a confidence (γ_v and γ_p) in that classification (lines 2-3 in Algorithm 2). The labels ω_v and ω_p are then converted to word embedding vectors (u_v and u_p) using the Word2Vec model (lines 4-5). At the same time, the set of company keywords K are also vectorized using Word2Vec, giving target vectors $k_1 \ldots k_m$ (also lines 4-5). Next, for each pair (u_*, k_i), where $*$ means v or p and $1 \leq i \leq m$, the cosine similarity is calculated (also lines 4-5):

$$\delta_*(u_*, k_i) = \sum_i \frac{u_* \cdot k_i}{\|u_*\|\|k_i\|}$$

Finally, an average score ν, weighted by confidences γ_v and γ_p is computed:

$$\nu = \frac{1}{Z}(\gamma_v \delta_v + \gamma_p \delta_p)$$

where Z is a normalizing constant that accounts for the variable size of K and the confidence values γ_v and γ_p (line 6). For each image x the score ν_x is the critic's estimate for how well the image communicates the keywords associated with company identity. For each pool j, the image x_j^* that has the highest score ν_j^* is selected for further exploration, and the vector z_j^* associated with that image is used as the location from which to continue exploring the generator's latent space. Because we used 100 pools, this results in 100 best (so far) images with associated z-vectors.

VGG16 The VGG16 model (Simonyan and Zisserman 2015) is a pre-trained convolutional neural network (CNN) model that won the image classification tasks in the Imagenet Large Scale Visual Recognition Challenge 2014 (ILSVRC2014). The model achieved 92.7% top-5 test accuracy on the ImageNet dataset (Deng et al. 2009), which contains over 14 million images and 1000 output classes. The model is freely available in frameworks like Pytorch or Keras and can be used off-the-shelf for many image classification tasks or it can be partially or completely fine-tuned on additional image sets. For our system, we did not perform additional fine-tuning. The model returns a probability distribution over its vocabulary, and we used this to compute the *maximum a posteriori* (MAP) estimate and took the corresponding vocabulary word as the output class and that word's probability under the distribution as the model's confidence.

Pythia Pythia is a framework created by Facebook Research, built on top of PyTorch, for vision and language research. It is openly available for solving challenges using vision and language datasets (Singh et al. 2018; 2019). Pythia is an excellent tool for recognizing details, answering queries about elements in images, and is different from regular image classifying algorithms, as it provides models that can "read" images. For our system, we used the query "What is shown in the image?" The most confident answer from Pythia was used as output, along with that answer's confidence.

Word2Vec Word2Vec (Mikolov et al. 2013) is an embedding model for text that transforms words into an n-dimensional vector representation. This vector representation acts as a set of abstract, distributed features that collectively "define" the word. The model is trained on

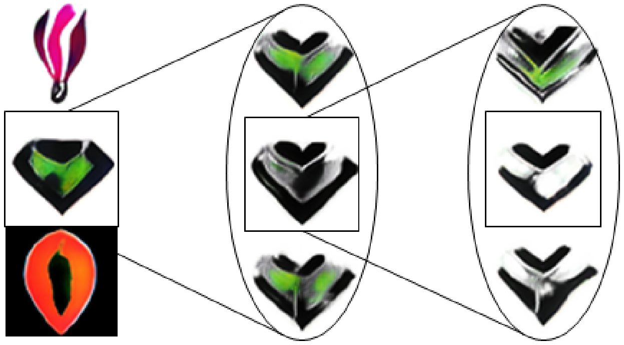


Figure 2: Visual example of the iterative progression of logos for the company T-Sprouts Babywear. The initial logos generated from a set of random z-vectors (left), after the first iteration (center) and final results (right).

word co-occurences in a large text corpus, with the loss designed to organize the space geometrically, such that similar vectors represent words with some semantic relationship and such that different geometric operations on vectors represent semantic operations on words [e.g., the now famous `vec`("king") - `vec`("man") + `vec`("woman") = `vec`("queen")]. This vector representation is useful because it can be used as input to deep neural networks. As it turns out, this feature representation also allows us to determine mathematically how related words are to each other based on their features, using, for example, the cosine similarity between vectors as a surrogate for the relatedness of the words to which those vectors are mapped.

Exploring z-space

Feedback from the critic is used to intelligently guide the search through the generator latent space by seeding the next round of generation with the z-vectors representing the current set of 100 best logos (one from each pool, lines 9-12 of Algorithm 1). This is done by computing 100 random perturbations of each z-vector, resulting in a new batch of 100 pools of 100 candidates (line 6), localized around the best z-vectors from the previous iteration. These new z-vectors are again used as input to the GAN to generate a new set of candidate images (line 7) which are again evaluated (line 8). This process is repeated until some stopping criterion is met (line 5). Finally, the last set of best images x_j^* is returned.

In our experiments, we observed significant improvement in image quality with only a couple of iterations beyond the initial batch generation, suggesting that the process may converge to a set of good suggestions fairly quickly in many cases (see Figure 2).

The returned set of final logos x_j^* can be further curated by human evaluation based on their appeal and relevance to the company. This final process could be considered analogous to a designer considering a collaborator's initial ideas or the final selection made by the company's executive board.

Company Name	Keywords
Sporting Goods	sports, outdoor, skiing, guns, climbing, camping
Big Bakery	basket, bread, pastry, food, dessert
Food Barn	food, barn, fruit, vegetables, milk
MH Clothing	hip, clothing, cheap, cheap, colorful
Juice Juice	juice, smoothie, drink, hangout, fun
Lu Lobster	lobster, fish, sea, ocean, lobster
Papa Pizza	john, papa, pizza, pepperoni, cheese
Rocky Mountain Bikes	mountain, bike, pump, road, rocky
Star Creme Coffee	coffee, swirl, cream, love, star
T-Sprouts Babywear	shirt, toddler, sprout, baby, toy

Table 1: Company names and target keywords use for logo generation experiments.

Results

To test our system, we invented ten fictional companies for which to create logos. We used fictional companies to avoid potential familiarity bias in the system (via data used for model training), during logo post-selection, and in survey respondents during external evaluation. Table 1 presents the company names and the related keywords used for generating the logos for each company. After the system produced its final batch of logo suggestions, we manually post-selected the logo we felt worked best given the company name and keywords, simulating the final say the company administration would have in the process. Figures 3-6 present the most (subjectively) appealing logos generated by the system for the respective companies.

The logo for Star Creme Coffee looks like a coffee cup viewed from above, with the suggestion of a 'C' inside. The logo for Juice Juice is a simple stroke 'J' on a bright yellow background. Both are (subjectively) quite appropriate (see Figure 3). Figures 4 and 5 show other logos created, some more successful than others: the Bakery logo is suggestive of rolls or loaves of bread; the Sporting Goods logo looks like a ball (perhaps tennis or basketball); the Papa Pizza logo is perhaps reminiscent of a pizza box; the other three look like logos of some kind, but it is more difficult to justify their relevance to the target companies, though the Lu Lobster logo's color is suggestive of lobster, the Food Barn logo could be a (grain) scoop and the MH Clothing logo could be initials.

Perhaps most striking are the logos generated for the companies Rocky Mountain Bikes and T-Sprouts Babywear, shown in Figure 6. The former looks like both a bike rider and a mountain range with the sun, with text at the bottom; the latter looks like a parent holding the hand of a toddler within a heart frame.

Some other interesting logos which give us more insight

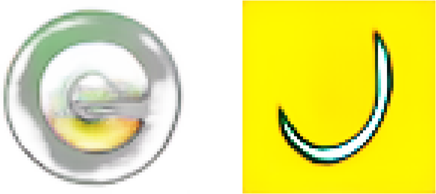


Figure 3: Best logos for Star Creme Coffee and Juice Juice

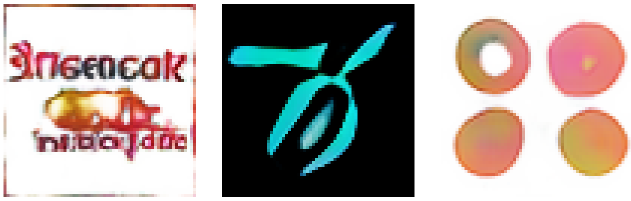


Figure 4: Best logos for Lu Lobster, Food Barn, and Bakery.

into how the scoring and evaluation work are shown in Figure 7. The left logo, for Lu Lobster, shows the text “FROG”, which has a high correlation (low distance) in Word2Vec with all the keywords mentioned in Table 1 (*Lobster, Fish, Sea and Ocean*). The right logo, for Star Creme Coffee, is suggestive of a star and the color scheme evokes the idea of a creamsicle.

Evaluation

In order to perform an external evaluation of the system’s effectiveness, we designed surveys to address the following questions:

1. Does exploring the GAN’s latent space using feedback from the evaluation module add value over traditional GAN generation?
2. Are the system-generated images good logos for their respective companies?

Efficacy of Latent Space Search

In order to test whether feedback-guided search of the latent space aided in the design process, we used the following experimental design. For each of the ten companies:

1. The system is run to generate a set A of n logos.
2. The set A goes through a final curation process by which the logo a deemed to best meet the company’s design requirements is chosen by hand (in essence, we simulated acting as the design team for the company, working with a collaborative CC system). The generated logos will be referred to as ReleLogos [short for “Rele(vant) Logos”] from here on.
3. The GAN of Mino and Spanakis is run to generate a set B of $5n$ logos.
4. The set B goes through a final curation process by which the logo b deemed to best meet the company’s design requirements is chosen by hand (again, simulating a design team for the company, with a different, less-collaborative, more tool-like system). These are called Random Logos.
5. A survey is conducted in which we ask, for each company, which logo, a or b (ReleLogo or Random Logo), is preferred. To avoid bias, we randomized the pair ordering for each question (see Figure 8).

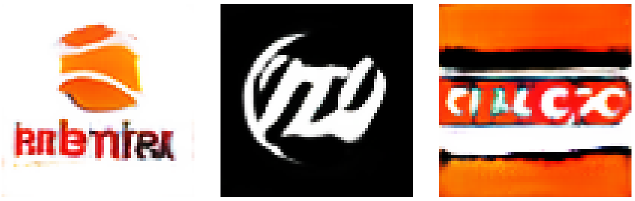


Figure 5: Best logos for Sporting Goods, MH Clothing and Papa Pizza.

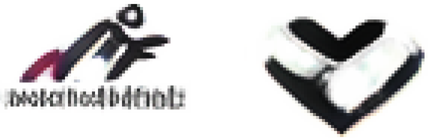


Figure 6: Best logos for Rocky Mountain Bikes and T-Sprouts Babywear

68 participants, from geographic locations in India, Nepal, the UK and the USA, responded to the survey, giving us 680 responses (68 for each of the 10 companies). Table 2 shows the total number of votes obtained for each of the companies for the ReleLogos and the Random Logos. A Paired t-test shows that the difference in average votes obtained for the ReleLogos and Random logos is big enough to be statistically significant ($p = 0.04011$).

The final row of the table shows that ReleLogo logos received nearly twice as many votes overall, and for 8/10 companies, our system’s logos were preferred by wide margins. Because both types of logos were post-selected by human “designers”, the fact that ReleLogos were chosen nearly 2:1 over the more random logos provides strong evidence that the feedback-based latent space search is providing significant value in the design process.

As an additional bit of anecdotal evidence for this claim, we note that in an informal survey of generated sets A and

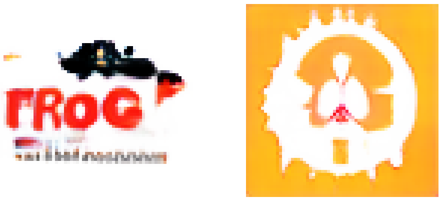
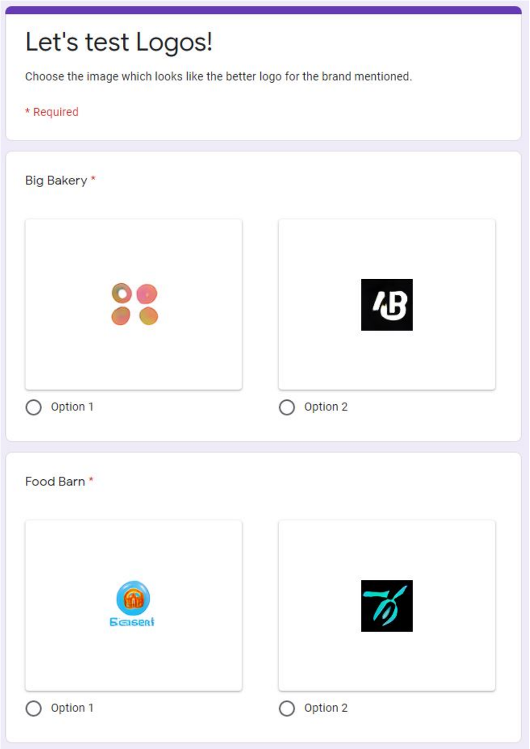


Figure 7: Interesting logos for Lu Lobster and Star Creme Coffee



Let's test Logos!

Choose the image which looks like the better logo for the brand mentioned.

* Required

Big Bakery *

○ Option 1 ○ Option 2

Food Barn *

○ Option 1 ○ Option 2

Figure 8: Survey 1 format. A pair of logos was presented for each of the 10 companies, with the order for each logo pair randomized.

B, it was observed that ≈10% of the logos generated by our system (set A) appeared to be relevant to company identity, whereas in the more randomly (LoGAN) generated dataset (set B), less than 1% were relevant in any way, suggesting a potentially $10\times$ improvement in efficiency of the design process.

Logo Quality

In order to evaluate how well the logos our system generates might be received as creative artifacts, we designed a second survey inspired by Jordanous's *evaluation guidelines* for computational creativity (2011). We identified four characteristics that might describe the goals for logo design: visual *coolness*, *relevance* to company identity, *intelligence* of design, and perceived *creativity*. We chose the seven logos we felt best demonstrate the capability of our system, and asked respondents to rate each of them for the four characteristics. Because these were not relative to any baseline, we settled on a 3-item Likert scale for evaluating each characteristic: *yes*, *maybe*, *no* (see Figure 9).

To avoid bias, the second survey involved new participants (20 total), and we targeted people with experience in computational creativity research.

The results are depicted in Figures 10-13. Most notably, the Rocky Mountain Bikes logo was clearly a success, with only a single *no* vote across all four criteria. In addition, 5/7 logos were pretty clearly considered at least somewhat cool looking; 6/7 were considered at least somewhat relevant to company identity; 4/7 were considered at least somewhat intelligent design ideas; and 5/7 were considered at least somewhat creative.

Company Name	Votes for ReleLogos	Votes for Random Logos
Big Bakery	31	37
Food Barn	17	51
Sporting Goods	42	26
Papa Pizza	58	10
Rocky Mountain Bikes	64	4
MH Clothing	42	26
Star Cream Coffee	55	13
Juice Juice	55	13
T-Sprouts Babywear	42	26
Total #votes	446	234

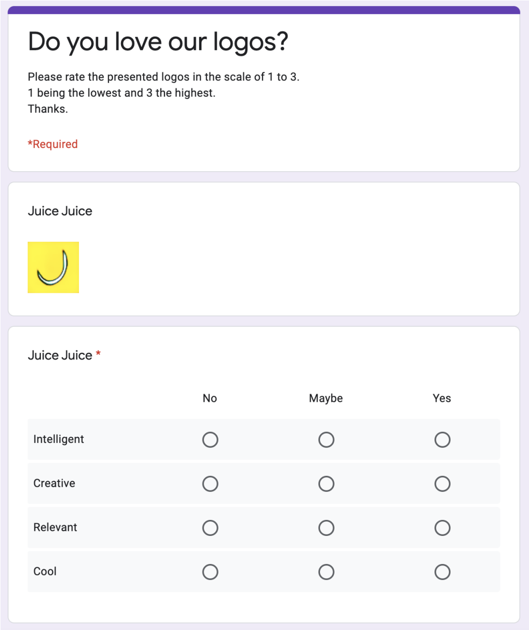
Table 2: Survey 1: Votes for ReleLogos versus Random Logos for each of ten fictional companies. ReleLogos was the winner in 8/10 cases, and a Paired t-test indicates that the preference difference between the two choices is statistically significant ($p = 0.04011$).

Table 3 is an agglomerated view of various averages across the 20 participants (Likert values were converted to numerical values as *no*: 0, *maybe*: 1, *yes*: 2). The first four rows show average (across companies) response rates for each of the characteristics—while there are clear wins for each of these at the company level, some less successful designs result in overall ambivalent averages; still, on average, the system does not fail for any of the characteristics. The next seven rows show average (across characteristics) response rates for each of the companies—here there are some clear standouts, both positive (Star Creme Coffee, T-Sprouts Babywear and Rocky Mountain Bikes) and negative (Lu Lobster and Papa Pizza). The last row is an overall average across both companies and characteristics—a somewhat encouraging maybe.

Discussion

The GAN-based approach of Sage et al. provides an interesting model and an end-to-end pipeline for logo generation. However, because it has no mechanism for conditioning the output, nor for the system to self-evaluate the resulting logos, it is of limited use for facilitating the process of logo design and selection for any particular company—the vast majority of the output logos will always be irrelevant to any specific designer or company, even though they will look generally like logos. Mino and Spanakis improve on this by allowing some conditioning (the designer can have some color control), but their improvements still result in far too many irrelevant outputs.

Building on their work, we have shown how a GAN-based



Do you love our logos?

Please rate the presented logos in the scale of 1 to 3.
1 being the lowest and 3 the highest.
Thanks.

*Required

Juice Juice

Juice Juice *

	No	Maybe	Yes
Intelligent	○	○	○
Creative	○	○	○
Relevant	○	○	○
Cool	○	○	○

Figure 9: Survey 2 format. Each company logo design is rated on four characteristics: *intelligence*, *coolness*, *relevance* and *creativity*.

generator can be informed by feedback from an evaluator module that takes into account linguistic cues (and could be expanded to include other kinds of relevant conditioning information). The result is intelligent search of the GAN's latent space, focusing the output in at least two ways: because images with greater visual recognizability are identified with more confidence by the vision modules, the system naturally gravitates to searching the latent space for images that are visually understandable; and images that are more relevant to the company (as measured by keyword similarity) get more preference and thus also bias the latent space search towards images that are relevant to the company. This combination of search pressures drives the GAN to produce images that are *recognizably relevant* to designer/company goals.

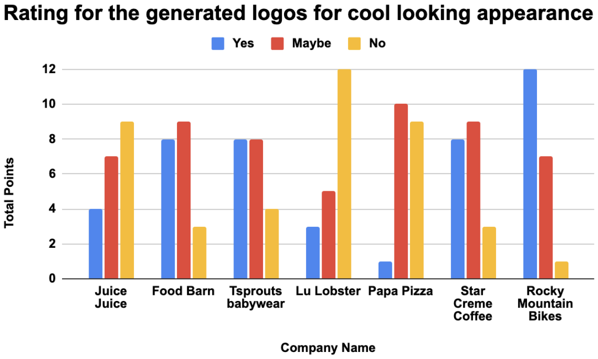


Figure 10: Ratings for *coolness* of the ReleLogos.

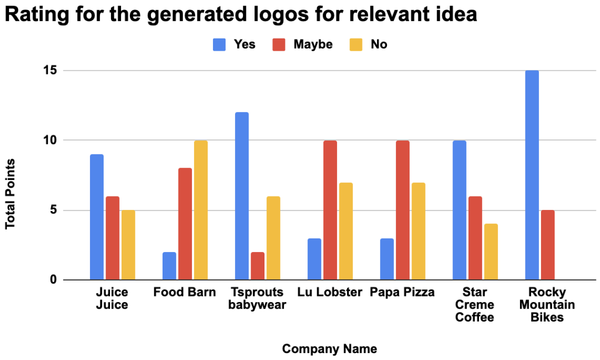


Figure 11: Ratings for *relevance* of the ReleLogos.

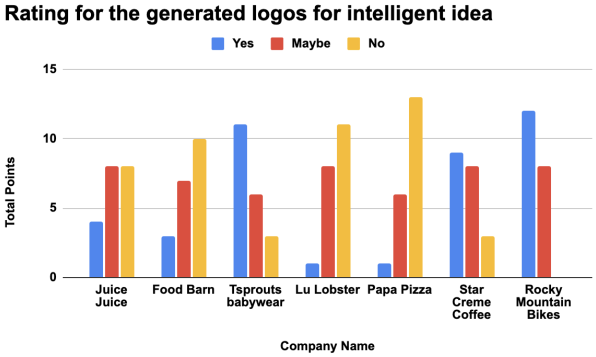


Figure 12: Ratings for *intelligence* of the ReleLogos.

The fact that the search for quality logo images is conducted in the GAN's latent space (rather than in the much more complex raw pixel space) both allows the process to be more efficient and to incorporate linguistic information (in the form of vectorized embeddings).

Conclusions and Future Work

This paper presents a novel methodology for generating and evaluating logos for a specific company or a brand. The benefit of this approach is the ability to intelligently search the latent space of the generator using feedback from visuo-linguistic evaluation, and the general approach should be immediately applicable to many other visual creation tasks. While our methods currently most naturally apply to visual tasks, with additional work it is possible that they may be further generalized, either by generalizing GAN models beyond visual generative tasks or by using other types of generative models (e.g., transformers) in their place.

While the system can be used autonomously, with full creative control, we envision it more as a collaborator, and, for now, best results are obtained with some human post-selection of the final system output.

Two different external evaluations verified that

1. intelligent search of the generator's latent space provides value over random generation, even given hu-

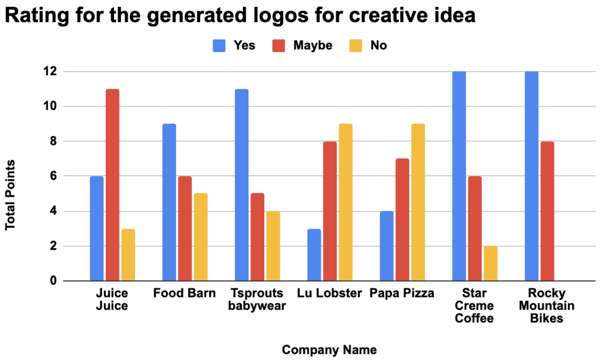


Figure 13: Ratings *creativity* of the ReleLogos.

Average rating type	ReleLogos
Intelligence	0.95
Creativity rating	1.17
Relevance rating	1.11
Coolness rating	1.02
Juice Juice rating	0.98
Food Barn rating	0.93
Star Creme Coffee rating	1.34
T-Sprouts Babywear rating	1.31
Lu Lobster	0.64
Papa Pizza	0.64
Rocky Mountain Bikes	1.63
All	1.064

Table 3: Survey 2: Average ratings across different viewpoints for ReleLogos. The system produces logos that, on average across characteristics, across companies, and across both, show encouraging potential.

man post-selection

2. the resulting logos are perceived as sometimes cool looking, reasonably intelligent ideas, and generally both relevant and creative

While these survey results are positive, they should be considered as encouraging preliminary indications that demand further validation with more rigorous and larger scale evaluation instruments.

The model does not yet deal well with text on the images. Pythia was queried for “What is written on the image?”, but it did not yield good results. For future work, it would be better to implement Optical Character Recognition (OCR) for text-based logos. Logos with better designs often have the initials or name of the company on them, so it would be a nice addition to focus on text-based logos, and including OCR algorithms would likely improve some designs as well as benefit the scoring mechanism. Increasing model power and image resolution would also likely improve final results.

Automation of the post-selection final step could be done, for example, by implementing a deep neural “aesthetic” network, which would make the process more convenient by requiring no human input, at the cost, of course, of allowing no human input. Conversely, the system could be made less autonomous by allowing more human input: humans could act as (additional) image critics or, if a z-vector encoder was available, human-created images could be used to stimulate the system’s latent space search (thanks to an anonymous reviewer for this suggestion). This tighter coupling with human input has the potential to make the system more co-creative by making it more *dialogic*, as suggested by Bown et al. (2020).

One interesting direction for exploration, given the current trends in deep learning, is the possibility of building an end-to-end trainable system—z-vector and conditioning information in, logos out—with the entire model learning in one pipeline. The biggest obstacle to this at the moment seems to be how to properly wrap the evaluation mechanism in a differentiable loss function. Because both Word2Vec and VGG16 are vector-based models that employ embeddings and are themselves trained using a loss function, there seems to be some hope that this could be done (Pythia may be more problematic, but other vision systems could also be considered). If such a loss function can be realized, this opens up the possibility of backpropagating loss through the vision system and then through the generator back to the initial z-vector input. In this way, the latent space could be searched using gradient descent rather than using the random perturbations we use now, likely resulting in significant improvement in both system run times and quality of output. The vision/classification and generating systems could be trained at the same time, or, they could still be used as off-the-shelf modules as has been done here.

References

Atarsaikhan, G.; Iwana, B. K.; and Uchida, S. 2018. Contained neural style transfer for decorated logo generation. In *Proceedings of the 13th IAPR International Workshop on Document Analysis Systems*, 317–322.

Bown, O.; Grace, K.; Bray, L.; and Ventura, D. 2020. Speculative exploration of the role of dialogic AI in co-creation. In *Proceedings of the 11th International Conference on Computational Creativity*, 25–32.

Cunha, J. M.; Martins, P.; Gonçalo, H.; and Machado, P. 2020. Ever-changing flags: Trend-driven symbols of identity. In *Proceedings of the 8th Conference on Computation, Communication, Aesthetics & X*, 215–228.

Cunha, J. M.; Martins, P.; and Machado, P. 2018. How shell and horn make a unicorn: Experimenting with visual blending in emoji. In *Proceedings of the Ninth International Conference on Computational Creativity*, 145–152.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27, 2672—2680.

Heath, D.; Norton, D.; and Ventura, D. 2014. Conveying semantics through visual metaphor. *ACM Transactions on Intelligent Systems and Technology* 5(2):article 31.

Jordanous, A. 2011. Evaluating evaluation: Assessing progress in computational creativity research. In *Proceedings of the Second International Conference on Computational Creativity*, 102–107.

Kingma, D. P., and Welling, M. 2014. Auto-encoding variational Bayes. In *Proceedings of the 2nd International Conference on Learning Representations*.

Li, Y.-N.; Zhang, K.; and Li, D.-J. 2017. Rule-based automatic generation of logo designs. *Leonardo* 50(2):177–181.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. In *Proceedings of the 1st International Conference on Learning Representations*.

Mino, A., and Spanakis, G. 2018. LoGAN: Generating logos with a generative adversarial neural network conditioned on color. In *Proceedings of the 17th IEEE International Conference on Machine Learning and Applications*, 965–970.

Özbal, G.; Pighin, D.; and Strapparava, C. 2013. BrainSUP: Brainstorming support for creative sentence generation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, 1446–1455.

Ritchie, G. 2007. Some empirical criteria for attributing creativity to a computer program. *Minds and Machines* 17:67–99.

Sage, A.; Agustsson, E.; Timofte, R.; and Van Gool, L. 2018. Logo synthesis and manipulation with clustered generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5879–5888.

Simonyan, K., and Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. In *Proceedings of the 3rd International Conference on Learning Representations*.

Singh, A.; Natarajan, V.; Jiang, Y.; Chen, X.; Shah, M.; Rohrbach, M.; Batra, D.; and Parikh, D. 2018. Pythia—A platform for vision & language research. In *Proceedings of the NeurIPS workshop on Systems for Machine Learning*.

Singh, A.; Natarajan, V.; Shah, M.; Jiang, Y.; Chen, X.; Batra, D.; Parikh, D.; and Rohrbach, M. 2019. Towards VQA models that can read. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8317–8326.

Tomašič, P.; Papa, G.; and Žnidaršič, M. 2015. Using a genetic algorithm to produce slogans. *Informatica* 39(2):125–133.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L. u.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 6000–6010.