Creative Generation of 3D Objects with Deep Learning and Innovation Engines

Joel Lehman
Center for Computer Games Research
IT University of Copenhagen
Copenhagen, Denmark
lehman.154@gmail.com

Sebastian Risi
Center for Computer Games Research
IT University of Copenhagen
Copenhagen, Denmark
sebr@itu.dk

Jeff Clune
Department of Computer Science
University of Wyoming
Laramie, Wyoming, USA
jeffclune@uwyo.edu

Abstract

Advances in supervised learning with deep neural networks have enabled robust *classification* in many real world domains. An interesting question is if such advances can also be leveraged effectively for computational *creativity*. One insight is that because evolutionary algorithms are free from strict requirements of mathematical smoothness, they can exploit powerful deep learning representations through arbitrary computational pipelines. In this way, deep networks trained on typical supervised tasks can be used as an ingredient in an evolutionary algorithm driven towards creativity. To highlight such potential, this paper creates novel 3D objects by leveraging feedback from a deep network trained only to recognize 2D images. This idea is tested by extending previous work with *Innovation Engines*, i.e. a principled combination of deep learning and evolutionary algorithms for computational creativity. The results of this automated process are interesting and recognizable 3D-printable objects, demonstrating the creative potential for combining evolutionary computation and deep learning in this way.

Introduction

There have recently been impressive advances in training deep neural networks (DNNs; Goodfellow, Bengio, and Courville 2016) through stochastic gradient descent (SGD). For example, such methods have led to significant advances on benchmark tasks such as automatic recognition of images and speech, sometimes matching human performance. (He et al. 2015). While impressive, such advances have generally been limited to *supervised* classification tasks in which a large number of labeled examples is available. Such a process cannot readily create interesting, unexpected outputs.

As a result, DNNs have not precipitated similar advances in *creative* domains. Creating new artifacts does not fit naturally into the paradigm of SGD, because (1) creativity often lacks a clear error signal and (2) creative systems are often non-differentiable as a whole, i.e. they may encompass arbitrary computation that lacks the mathematical smoothness necessary for SGD. Combining evolutionary algorithms (EAs) with DNNs can remedy both issues. One powerful such combination is explored in this paper: The latent knowledge of the DNN can be leveraged as a reward signal for an EA; and evaluation in an EA can freely incorporate arbitrary computation.

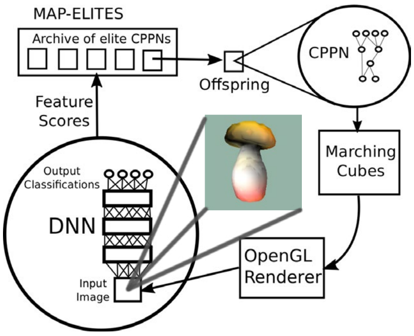


Figure 1: **Approach.** Each iteration a new offspring CPPN is generated, which is used to generate a 3D vector field. The marching cubes algorithm extracts a 3D model from this field, which is rendered from several perspectives. The rendered images are input into an image-recognition DNN, and its output confidences supply selection signals for MAP-Elites, thereby driving evolution to find 3D objects that the DNN increasingly cannot distinguish from real-world ones.

Building upon Nguyen, Yosinski, and Clune (2015b), the main idea in this paper is that classification DNNs can be coupled with creative EAs to enable *cross-modal* content creation, where a DNN's knowledge of one modality (e.g. 2D images) is exploited to create content in another modality (e.g. 3D objects). While the EA in Nguyen, Yosinski, and Clune (2015b) created images from an image-based DNN, SGD can also do so (Yosinski et al. 2015). In contrast, the system described here creates *3D models* from a *2D image* recognition DNN, making use of a non-differentiable rendering engine and an expressive evolutionary computation (EC) representation called a compositional pattern-producing network (CPPN; Stanley 2007). In this way, the unique advantages of both EAs and DNNs are combined: EAs can leverage compressed genetic representations and evaluate individuals in flexible ways (e.g. bridging information modalities), while DNNs can create high-level object representations as a byproduct of supervised training.

The paper's approach (Figure 1) combines an image-recognition DNN with a diversity-generating EA (MAP-Elites; Mouret and Clune 2015), to realize an *Innovation Engine* (Nguyen, Yosinski, and Clune 2015b). The extension here is that the genetic encoding of EndlessForms.com (Clune and Lipson 2011) enables automatic sculpting of 3D models. In particular, evolved objects are rendered with a 3D engine from multiple perspectives, resulting in 2D images that can be evaluated by the DNN. The classification outputs of the DNN can then serve as selection pressure for MAP-Elites to guide search. In this way, it is possible to create novel 3D models from a DNN, enabling creative synthesis of new models from only labeled images.

The product of running this system is a collection of 3D objects from $1,000$ categories (e.g. banana, basketball, bubble), many of which are indistinguishable to the DNN from real-world objects. A chosen set of objects is then 3D printed, showing the possibility of automatic production of novel real-world objects. Further, a user study of evolved artifacts demonstrates that there is a link between DNN confidence and human recognizability. In this way, the results reveal that Innovation Engines can automate the creation of interesting and often recognizable 3D objects.

Background

The next section first reviews previous approaches to creative generation of objects. After that, deep learning and MAP-Elites are described; together they form a concrete implementation of the Innovation Engine approach, which is applied in this paper's experiments and is reviewed last.

Creative Object Generation

EndlessForms.com (Clune and Lipson 2011) is a collaborative interactive evolution website similar to Picbreeder.org (Secretan et al. 2008), but where users collaborate to evolve diverse 3D objects instead of 2D images. Using the same genetic encoding, this paper attempts to automate the creativity of EndlessForms.com users, similarly to how Innovation Engines were originally applied (Nguyen, Yosinski, and Clune 2015b) to automate the human-powered collaborative evolution in Picbreeder (Secretan et al. 2008). Importantly, this work builds upon previous approaches that exploit combinations of ANNs and EC (Baluja, Pomerleau, and Jochem 1994) or classifiers and EC (Correia et al. 2013; Machado, Correia, and Romero 2012) for automatic pattern-generation. Other similar approaches have applied EAs in directed ways to evolve objects with particular functionalities, like tables, heat-sinks, or boat hulls (Bentley 1996).

Shape grammars are another approach to generating models (Stiny and Gips 1971), where iteratively-applied grammatical rules enable automatic creation of models of a particular family. However, such grammars are often specific to particular domains, and require human knowledge to create and apply. The procedural modeling community also explores methods to automatically generate interesting geometries (Yumer et al. 2015), although such approaches are also subject to similar constraints as shape grammars.

Perhaps the most similar approach is that of Horn et al. (2015), where a user-supplied image is analyzed through four metrics, and a vase is shaped through an EA to match such characteristics. Interestingly, the approach here could be adapted in a similar direction to create sculptures inspired by user-provided images (by matching DNN-identified features instead of hand-designed ones), or even to create novel sculptures in the style of famous artists or sculptors (Gatys, Ecker, and Bethge 2015); such possibilities are described in greater detail in the discussion section.

Deep Learning

Although the idea of training multi-layer neural networks through back-propagation of error is not new, advances in computational power, in the availability of data, and in the understanding of many-layered ANNs, have culminated in a high-interest field called *deep learning* (Goodfellow, Bengio, and Courville 2016). The basic idea is to train many-layered (deep) neural networks on big data through SGD.

Deep learning approaches now achieve cutting-edge performance in diverse benchmarks, including image, speech, and video recognition; natural language processing; and machine translation (Goodfellow, Bengio, and Courville 2016). Such techniques are generally most effective when the task is *supervised*, i.e. the objective is to learn a mapping between given inputs and outputs, and when training data is ample. Importantly, the output of the DNN (and the error signal) must be composed only from differentiable operations.

One focus of deep learning is object recognition, for which the main benchmark is the ImageNet dataset (Deng et al. 2009). ImageNet is composed of millions of images, labeled from $1,000$ categories spanning diverse real-world objects, structures, and animals. DNNs trained on ImageNet are beginning to exceed human levels of performance (He et al. 2015), and the learned feature representations of such DNNs have proved useful when applied to other image comprehension tasks (Razavian et al. 2014). In this paper, DNNs are applied to sculpt 3D objects by providing feedback to the MAP-Elites EA, which is described next.

MAP-Elites

While most EAs are applied as optimization algorithms, there are also EAs driven instead to collect a wide *diversity* of high-quality solutions (Pugh et al. 2015; Laumanns et al. 2002; Saunders and Gero 2001). Because of their drive towards diverse novelty, such algorithms better fit the goals of computational creativity than EAs with singular fixed goals.

One simple and effective such algorithm is the multi-dimensional archive of phenotypic elites (MAP-Elites) algorithm (Mouret and Clune 2015), which is designed to return the highest-performing solution for each point in a space of user-defined feature dimensions (e.g. the fastest robot for each combination of different heights and weights). The idea is to instantiate a large space of inter-related problems ($1,000$ in this paper), and use the current-best solutions for each problem as stepping stones to reach better solutions for any of the other ones. That is, solutions to easier problems may aid in solving more complex ones. Note that only a cursory description of MAP-Elites is provided here; Mouret and Clune (2015) provides a complete introduction.

MAP-Elites requires a domain-specific measure of performance, and a mapping between solutions and the feature space. For example, if the aim is to evolve a variety of different-sized spherical objects, the performance measure could be a measure of roundness, while the feature space dimension could be object size. In this way, MAP-Elites has a mechanism to separate the quality criterion (e.g. roundness) from dimension(s) of desired variation (e.g. size). In practice, because the feature space is often continuous, it is first discretized into a finite set of *niches*.

A map of elite solutions is then constructed, that maintains the current elite solution and its corresponding performance score for each niche. When a new solution is evaluated, it is mapped to its niche, and its performance is compared to that of its niche's current elite. If the newly-evaluated solution scores higher than the the old elite individual, it replaces the old elite in the niche, and the niche's score is updated accordingly.

Evolution is initialized with an empty map, which is seeded by evaluating a fixed number of random solutions. A fixed budget of evaluations is then expended by repeatedly choosing a solution at random from the map, mutating it, and then evaluating it. After all evaluations have been exhausted, the final map is returned, which is the collection of the best solution found in each niche.

Innovation Engines

The MAP-Elites algorithm described above can be used to realize an Innovation Engine (Nguyen, Yosinski, and Clune 2015b). Like the DeLeNoX approach (Liapis et al. 2013), Innovation Engines combine (1) EAs that can generate and collect diverse novelty with (2) DNNs that are trained to distinguish novelty and evaluate its quality. The hope is that such an architecture can produce a stream of interesting creative artifacts in whatever domain it is applied to.

This paper builds on the initial implementation in Nguyen, Yosinski, and Clune (2015b), where a pretrained image recognition DNN is combined with MAP-Elites to automatically evolve human-recognizable images. In that work, the space of MAP-Elites niches was defined as the $1,000$ object categories within the ImageNet dataset (Deng et al. 2009), which is a common deep learning benchmark task (note that the same space of niches is applied here). CPPNs that represent images (as in Picbreeder; Secretan et al. 2007) were evolved, and the performance measure for MAP-Elites was to maximize the DNN's confidence that an evolved image is of a specific object category. An evolutionary run thus produced a collection of novel images, many of which resembled creative depictions of real-world objects. The work was not only accepted into a competitive university art show, but won an award (Nguyen, Yosinski, and Clune 2015b). The work here expands upon such image evolution, applying a similar technique to evolve 3D objects.

Note that the current version of Innovation Engines can be seen in Boden's terminology as realizing *exploratory creativity* but not *transformational creativity* (Boden 1996). That is, while the algorithm has a broadly expressive space of images or objects to search through, its conception of what objects are interesting and why they are interesting is fixed. In the future, unsupervised deep learning may provide a mechanism to extend innovation engines with aspects of transformational creativity (Nguyen, Yosinski, and Clune 2015b); for example, the DeLeNoX system uses unsupervised autoencoder neural networks to iteratively transform its creative space (Liapis et al. 2013).

Approach

While ideally advances in deep learning would also benefit computational creativity, creative domains often encompass arbitrary computation and reward signals that are not easily combined with the gradient descent algorithm. The approach here is thus motivated by the insight that EAs, unlike DNNs, are not limited to pipelines of computation in which each stage is differentiable. In particular, one interesting possibility enabled by EAs is to exploit the latent knowledge of the DNN to create structures with entirely different modality than with which the DNN's was trained.

For example, it is not clear how SGD can extract 3D objects from an image-recognition network, because there is no natural differentiable mechanism to translate from a 3D representation of an object to the 2D pixel representation used by image-recognition DNNs. In contrast, EC representations of 3D objects can be rendered to 2D pixel images through non-differentiable rendering engines; and the resulting images can interface with trained image-recognition DNNs. While it might be possible to train a 3D object recognition DNN (e.g. with necessary technical advances and an appropriate dataset), there are diverse cross-modal possibilities that EAs enable (particular examples can be found in the discussion section). In other words, this idea provides a general mechanism for creative cross-modal linkage between EAs and DNNs, which respects the advantages of both methods: EAs do not require differentiability, while DNNs better leverage big data and computational efficiency to learn powerful hierarchies of abstract features.

This paper realizes a proof-of-concept of cross-modal linkage, shown in Figure 1, wherein 3D objects are represented with the EndlessForms.com encoding (Clune and Lipson 2011). This encoding represents a mapping from 3D coordinate space to material density, by using a CPPN (which is similar to a neural network function approximator). Inspired by regularities of biological organisms, activation functions in such CPPNs are drawn from a set chosen to reflect such regularities, thereby enabling representing complex patterns compactly.

In more detail, the CPPN takes as input Cartesian coordinates and generates as its output the density of material to be placed in that coordinate. The CPPN is then queried systematically across the 3D coordinate space, resulting in a 3D scalar field. Next, the marching cubes algorithm (Lorensen and Cline 1987) constructs a mesh that wraps the scalar field, by defining the object's boundary as a threshold of material density. Note that the EndlessForms.com encoding is extended here to enable more detailed models that vary in color across their surface. To accomplish this effect, outputs are added to the CPPN that specify the HSV color of each voxel, enabling the creation of objects with detailed colors.

To evaluate an individual, this encoding is combined with a rendering engine that produces several rendered images of the encoded object from different perspectives. Then these rendered images are input into a pretrained DNN to produce performance signals for the MAP-Elites EA. The chosen DNN is the BLVC reference GoogleNet from the Caffe model zoo (Jia et al. 2014), a freely-available DNN similar in architecture to GoogLeNet (Szegedy et al. 2015).

As in Nguyen, Yosinski, and Clune (2015b), the underlying MAP-Elites algorithm's space of niches is defined by the $1,000$ discrete object classes that the DNN can recognize (which span household objects, animals, and vehicles). Performance for each niche is defined as the DNN's confidence that the generated artifact is an example of the class the niche represents. In particular, the confidences of the six renders for each class are multiplied together; this was shown in preliminary experiments to improve performance.

Rendering Improvements

To improve the render quality of the 3D objects, two additions to the algorithm are considered: (1) enabling lighting and material properties of the object to evolve, and (2) enabling the background color to evolve. Overall, rendering quality is important because the DNN is trained on real-world photographs, and therefore may rely on features such as lighting or background context to discriminate between objects. As a result, the success of the approach may depend on the kinds of images that are possible or easy to represent.

For this reason, in addition to the CPPN, the genome has four evolvable floating-point numbers that encode parameters of lighting (the diffuse and ambient intensities) and the object's material (its shininess and its specular component); and three evolvable parameters that encode the HSV of the background color. All such parameters have a fixed chance of mutating when a new offspring is produced.

These extensions enable evolution to control aspects of a rendered image unrelated to evolving a 3D object. For example, adjusting the background color can help control for a superficial discriminative feature that may always correlate with the presence of certain objects. For example, fish may nearly always be found in the context of water, and so a DNN may only recognize an evolved object as a fish if it is rendered against a blue background.

Search Improvements

To improve the effectiveness of the underlying MAP-Elites search process, two additions are considered: (1) adding niches that represent more general classes of objects to enable incremental learning; and (2) biasing search away from exploring niches that produce fewer innovations.

Previous work found it was difficult to evolve images for very specific classes (e.g. nuanced breeds of dogs; Nguyen, Yosinski, and Clune 2015b), which preliminary experiments confirmed was also problematic for evolving 3D objects. Because of how supervised training of DNNs generally works, the $1,000$ target categories in ImageNet only represent the finest level of granularity. That is, SGD works most easily when an image is associated with only one label, even when broader categorical information is available. Because the niches for MAP-Elites are directly imported from the DNN's target categories, the EA must directly evolve towards specific nuanced categories. However, learning general concepts, e.g. distinguishing a dog from other animals, often provides scaffolding for learning more nuanced ones, e.g. distinguishing a pug from a french bulldog.

Thus, the idea is to artificially create more general niches by aggregating the specific categories together. Because the WordNet database (Miller 1995) underlies the categories of images in ImageNet, hierarchical relations in WordNet can be leveraged to cluster semantically similar categories. In particular, a tree is constructed consisting of all the ImageNet categories, with directed edges added for each hypernym relationship, from the more specific class to the more general one. Non-leaf nodes in this graph thus represent increasingly general concepts, which can be added to MAP-Elites to augment the more specific ones. Given the classification outputs of the DNN for a particular rendered image, the score for any of the added niches is calculated for by summing the confidences of all the leaf nodes beneath it, i.e. all the hyponym nodes. Because individuals can be maintained that maximize general concepts, additional pathways for incremental learning are enabled for evolution.

The second addition biases MAP-Elites away from expending resources on niches that have proved unproductive. In particular, each MAP-Elites niche is augmented with a decrementing counter. Each counter is initialized to a fixed value (10 in the experiments here) that determines the relative probability of choosing that niche for reproduction.

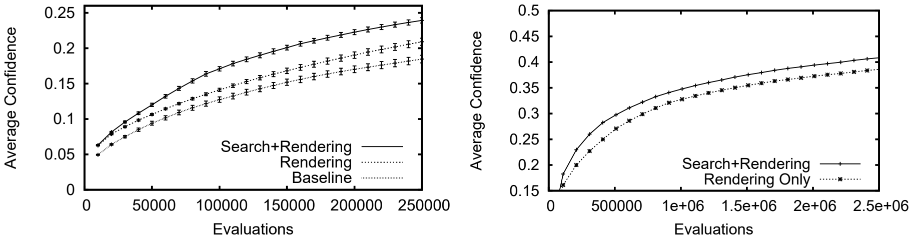
A niche's counter is decremented when an offspring generated from the niche's current champion does not replace any existing individuals in the map of elites (i.e. the niche is penalized because it did not lead to an innovation). If instead the offspring displaces other champions, then the counters for the initial niche and the niches of all displaced champions are reset to their initial maximum value.

Experimental Setup

The basic setup is replicated from Nguyen, Yosinski, and Clune (2015b), wherein the MAP-Elites algorithm is driven by the classification outputs of a DNN. However, the DNN here processes several renderings of a *3D object* instead of a single image. In particular, the object is rendered six times, successively rotated by 45 degrees increments around its y-axis (yaw). The motivation is to encourage the evolution of objects that resemble the desired class when viewed head-on from a variety of perspectives. Every alternating rendering is also rotated 5 degrees around its x-axis (pitch), to encourage further robustness. The voxel field for the EndlessForms.com encoding is given a resolution of 20 x 20 x 20 units, striking a balance between possible model detail and the computational cost of querying the CPPN for each voxel. Full source code, experimental results (including downloadable model files, and renders from all six evaluated perspectives), and user study data are freely available from the project website: `http://jal278.github.io/iccc2016/`.

Ablation Experiments

One practical concern is that evaluation of an individual is expensive computationally, as it requires (1) querying a CPPN $8,000$ times to generate the $20 \times 20 \times 20$ scalar field from which marching cubes produces a model, (2) rendering an image of that resulting model multiple times (here, 6), and (3) evaluating the rendered images with a large DNN. The DNN evaluation in particular is the computational bottleneck and depends upon capable GPUs to be time-efficient.



(a) Ablation algorithms

(b) Long runs

Figure 2: **Performance of evolutionary runs.** (a) The performance of incremental ablations of the proposed algorithm is shown averaged over ten independent runs. Performance is measured by averaging the confidence of the DNN first over all renderings of a particular object, and then over all object classes. All distinctions between methods are significant ($p < 0.05$), highlighting that the improvements both to lighting and to the search algorithm facilitate evolving objects that the DNN more confidently classifies. Note that error bars reflect standard error. (b) The performance over evaluations of the two long runs is shown, anecdotally highlighting how including both search and rendering additions appears also to result in long-term improvements. Many niches evolve towards high-confidence (18.2% of niches in *S+R* and 15.1% of niches in *R only* achieved confidence scores > 0.85 when averaged over object renderings). However, evolution in niches representing highly-specific objects (e.g. one breed of dog) or geometrically-complex ones (e.g. a grocery store) often stagnates.

As a result, performing a single run to highlight the full potential of the system (2.5 million evaluations) took around three weeks on the hardware available to the experimenters (a modern gaming laptop with a high-quality GPU).

To inform the longer runs, a series of shorter ablation runs ($250,000$ evaluations each) were first performed to validate algorithmic features. In particular, three algorithms were compared: a baseline (control) algorithm, the same algorithm augmented with the rendering improvements, and an algorithm with both the rendering and search improvements. The idea is to examine whether the added features result in better performance, thereby providing guidance for what algorithm should be applied when generating the main results.

Ablation Results

The results of the comparison are shown in Figure 2(a). The order of final performance levels achieved by the algorithmic variants reflects that adding the tested components significantly improves performance ($p < 0.05$; all tests are Mann-Whitney U-tests unless otherwise specified).

Both the rendering and search improvements comprise multiple sub-improvements; to better understand each component's relative contribution, shorter ablation experiments were conducted (10 independent runs of $70,000$ evaluations each). For search improvements, pruning unproductive niches provided greater benefit than did only including more general niches ($p < 0.05$), but both improved performance over the control algorithm ($p < 0.05$). For rendering improvements, allowing the background color to evolve significantly increased performance, while allowing lighting to evolve did not ($p < 0.05$). However, because they do not decrease performance, and because preliminary experiments revealed that lighting enabled more interesting aesthetic effects, lighting changes are included in the full experiments.

Main Experimental Results

Informed by the ablation experiments, two long runs were conducted (2.5 million evaluations). To verify anecdotally that the conclusions from the ablation experiments are likely to generalize to such longer runs, one run, called *S+R*, included the full suite of improvements (i.e. both search and rendering), while the other run, *R only*, included only the rendering improvements. The gross performance characteristics of the long runs are shown in Figure 2(b), and suggest that the algorithmic additions result in performance gains that persist even over runs with many more evaluations.

A curated gallery of high-confidence evolved objects is shown in Figure 3. To highlight the quality of learned object representations, mutations of selected objects are shown in Figure 4. Overall, the objects exhibit an interesting diversity and in most cases the connection between the object and the real world object class is evident.

3D Printing the Automatically Generated Objects

Because the output of the creative process are textured 3D models, it is possible to bring them into reality through 3D printing. A small selection of evolved objects was chosen from the results of both runs. In particular, objects were chosen that (1) were possible to print, (2) were colorful, and (3) highlighted interesting features used by DNNs to classify images. Model files were uploaded to the Shapeways commercial 3D printing service to be printed in their color sandstone material. The results of this process are shown in Figure 5. Note that many of the objects were too fragile to directly be printed (because their structures were too thin in particular areas). However, optimization criteria could be injected into the search process to mitigate such fragility.

To test the fidelity of the 3D printing process, the above photographs (without inlays) were also input to the training DNN, and the resulting highest-scored categories were recorded. The evolved Starfish was correctly classified by the DNN's first choice, while the Mushroom was classified first as a bolete (a type of wild mushroom), and the sixth ranked choice was the broader (correct) mushroom class. The Jellyfish was classified by first choice as a conch, and as jellyfish by the network's fifth choice. The Hammerhead was interestingly classified first as a *hammer*, and as eighth choice by the true hammerhead shark label. Finally, the Goldfinch was classified by fifth choice as a lorikeet, and as ninth choice a hummingbird; both also are colorful birds.

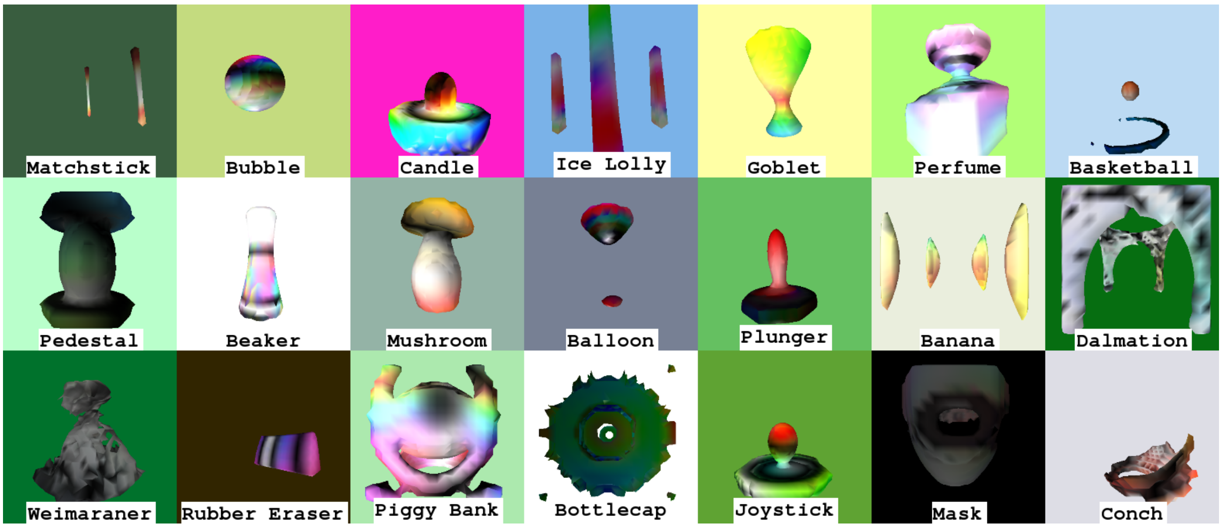


Figure 3: **Gallery of automatically generated high-confidence objects.** A curated selection of high-confidence champions from both the *S+R* and *R only* runs. Representing multiple copies of the same object (e.g. Banana, Ice Lolly, and Matchstick) helps maximize DNN confidence. The system often evolves roughly rotationally-symmetric objects (e.g. Goblet, Joystick, Bubble), both because many classes of real-world objects are symmetric in such a way and because it is the easiest way to maximize DNN confidence from all rendered perspectives. However, objects such as Conch, Mask, and Dalmatian show that asymmetric and more complex geometries can also evolve when necessary to maximize DNN confidence. Overall, the results show the promise of Innovation Engines for cross-modal creativity. Best viewed in color.

The conclusion is that even after crossing the reality gap, key features of objects are still be recognized by the DNN.

User Study

A user study was conducted to explore whether humans saw the resemblance of the evolved objects to their respective categories. In particular, 20 fixed survey questions were created by sampling from niches within which evolution successfully evolved high-confidence objects.

Each question asked the user to rank three images by their resemblance to the sampled category. One image was a rendering of an evolved object classified by the DNN with *high confidence* (i.e. the highest-confidence rendering for the intended category; always > 0.95). A second image was a rendering classified with *moderate confidence* (i.e. the rendering with score closest to 0.2, which is still qualitatively distinguished from the base expectation of 0.001). The third image was of an evolved object classified with high confidence as belonging to an arbitrarily-chosen *distractor* category. The idea is to see whether user rankings of the objects' resemblance to the true class agree with the DNN's ranking (i.e. high confidence, moderate confidence, distractor).

Twenty-three subjects were recruited using a social media post to fill out an online survey; the order of questions was fixed, but the order of images within each question was randomized. Users generally ranked images in an order similar to that of the DNN (Figure 6); the conclusion is that high-confidence objects generally bear semantic resemblance to the category they are optimized to imitate.

Discussion

The basic framework presented here could be used with DNNs trained on other image datasets to generate distinct types of 3D objects and scenes. For example, combining a DNN trained on the Places dataset (Zhou et al. 2014) with Google's deep dream visualization (Mordvintsev, Olah, and Tyka 2015) resulted in images of fantastical architectures, highlighting the potential for architectural creativity embedded in such a DNN. Thus substituting this paper's approach for deep dream may likewise yield interesting *3D* architectural creations. Similarly, DNNs trained to recognize other things could be leveraged to create diverse artifact types, e.g. 3D flowers, cars, or faces (pretrained DNNs for each such type of data are available from the Caffe model zoo).

The approach in this paper could also generalize to other kinds of cross-modal creation through non-differentiable computation. For example, a DNN trained to distinguish different speakers (Lee et al. 2009) could be leveraged to evolve parameters for speech synthesis engines, potentially resulting in diverse but realistic settings for speech synthesizers without human tuning. Another possibility is automatic creation of music; just as optimizing CPPN-based images led to more qualitatively interesting results than did optimization of a naive pixel-based representation (Nguyen, Yosinski, and Clune 2015b), optimizing CPPN-based representations of music (Hoover and Stanley 2009) fed through a music-recognition DNN (Lee et al. 2009) might similarly enable automatic generation of compositions with more coherent or interesting structure.

One possibility to enable more open-ended creativity would be to leverage high-level features encoded by the DNN to guide search, instead of only the classification labels. The novelty search algorithm (Lehman and Stanley 2011) could be applied to create objects that span the space of high-level representations. Because the features compos-

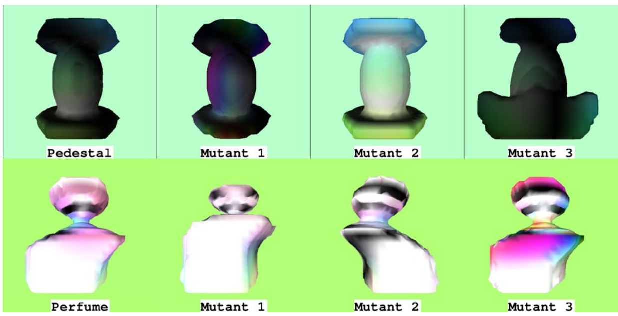


Figure 4: **Accessible variation from evolved objects.** First, twenty mutants of the Pedestal (top) and Perfume (bottom) champions were generated. For both rows, the original model is shown on the far left, followed to its right by three examples of interesting mutants. The conclusion is that in addition to the final objects, evolution also produces interesting object *representations* that can be leveraged for further creative purposes, e.g. interactive evolution.

ing the representation are constrained by their relation to classifying objects, exploration of such a space may yield a diversity of interesting objects. Conversely, the system could also be made more *directed* in interesting or interactive ways. For example, a novel 3D object might be optimized to mimic the high-level DNN features (Razavian et al. 2014) of a user-provided image, creating possibilities for human-machine artistic collaboration. Interestingly, such an approach could additionally be combined with the StyleNet objective function (Gatys, Ecker, and Bethge 2015) to sculpt objects inspired by a photograph, and cast in the *style* of a separate artwork or sculpture.

Finally, while created for computational creativity, the approach may also have implications for deployed deep learning systems. Nguyen, Yosinski, and Clune (2015a) suggested that DNNs may easily be fooled, given complete control of how an image is presented to the DNN. However, real world recognition systems may employ many (potentially unknown/unseen) cameras, which may preclude directly fooling such a system with a generated image. However, because evolved objects can be 3D printed, and because evolved objects are often recognized by diverse DNNs (data not shown), it may be possible to confound real-world deep learning recognition systems with such printed artifacts, even those based on multiple unseen cameras.

Conclusion

This paper introduced a framework for exploiting deep neural networks to enable creativity across modalities and input representations. Results from evolving 3D objects through feedback from an image recognition DNN demonstrate the viability of the approach: A wide variety of stylized, novel 3D models were generated that humans could recognize. The conclusion is that combining EC and deep learning in this way provides new possibilities for creative generation of meaningful and novel content from large labeled datasets.

Acknowledgements

Jeff Clune was supported by an NSF CAREER award (CAREER: 1453549).

References

Baluja, S.; Pomerleau, D.; and Jochem, T. 1994. Towards automated artificial evolution for computer-generated images. *Connection Science* 6(2-3):325–354.

Bentley, P. J. 1996. *Generic evolutionary design of solid objects using a genetic algorithm*. Ph.D. Dissertation, The University of Huddersfield.

Boden, M. A. 1996. *Dimensions of creativity*. MIT Press.

Clune, J., and Lipson, H. 2011. Evolving three-dimensional objects with a generative encoding inspired by developmental biology. *Proceedings of the European Conference on Artificial Life, See http://EndlessForms. com* 144–148.

Correia, J.; Machado, P.; Romero, J.; and Carballal, A. 2013. Evolving figurative images using expression-based evolutionary art. In *Proceedings of the fourth International Conference on Computational Creativity (ICCC)*, 24–31.

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

Gatys, L. A.; Ecker, A. S.; and Bethge, M. 2015. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*.

Goodfellow, I.; Bengio, Y.; and Courville, A. 2016. Deep learning. Book in preparation for MIT Press.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*.

Hoover, A. K., and Stanley, K. O. 2009. Exploiting functional relationships in musical composition. *Connection Science* 21(2-3):227–251.

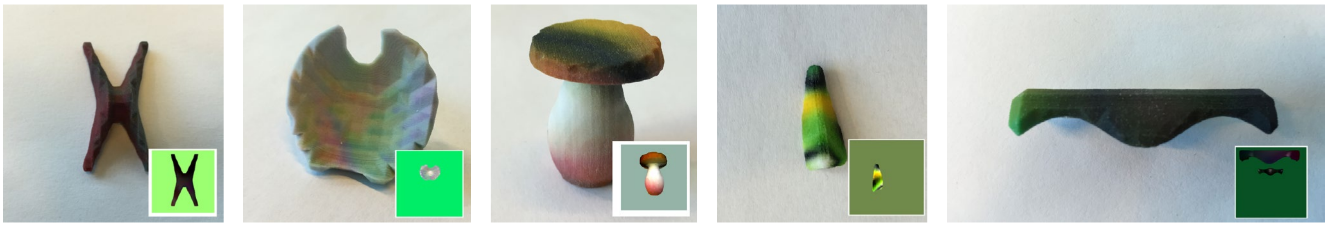
Horn, B.; Smith, G.; Masri, R.; and Stone, J. 2015. Visual information vases: Towards a framework for transmedia creative inspiration. In *Proceedings of the Sixth International Conference on Computational Creativity June*, 182.

Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R. B.; Guadarrama, S.; and Darrell, T. 2014. Caffe: Convolutional architecture for fast feature embedding. In *ACM Multimedia*, volume 2, 4.

Laumanns, M.; Thiele, L.; Deb, K.; and Zitzler, E. 2002. Combining convergence and diversity in evolutionary multiobjective optimization. *Evolutionary computation* 10(3):263–282.

Lee, H.; Pham, P.; Largman, Y.; and Ng, A. Y. 2009. Unsupervised feature learning for audio classification using convolutional deep belief networks. In *Advances in neural information processing systems*, 1096–1104.

Lehman, J., and Stanley, K. O. 2011. Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary computation* 19(2):189–223.



(a) Starfish (b) Jellyfish (c) Mushroom (d) Goldfinch (e) Hammerhead

Figure 5: **Gallery of 3D printed objects.** Shown are photographs of 3D-printed objects oriented similarly to one of their simulated renders (which is inlaid). Note that the evolved Hammerhead consisted of two similar objects; one was chosen arbitrarily for 3D printing. In all cases, a clear resemblance is seen between each 3D-printed object and its render, demonstrating the feasibility of automatically generating real-world objects using the approach.

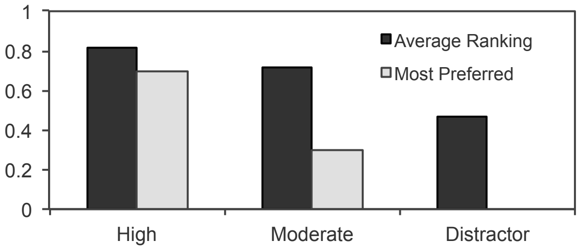


Figure 6: **User Study Results.** Black: The average rankings (normalized between 0 and 1) of each type of image (see text), aggregated across users and then questions. High and moderate confidence images are ranked significantly higher than distractor images ($p < 0.01$; Mann-Whitney U test). White: the percentage of questions in which users collectively ranked the image as *most* similar to the prompted category. Across all questions, the distractor is never the one that is most preferred across users. The conclusion is that user rankings often agree with the DNN.

Liapis, A.; Martınez, H. P.; Togelius, J.; and Yannakakis, G. N. 2013. Transforming exploratory creativity with DeLeNoX. In *Proceedings of the Fourth International Conference on Computational Creativity*, 56–63. AAAI Press.

Lorensen, W. E., and Cline, H. E. 1987. Marching cubes: A high resolution 3D surface construction algorithm. In *ACM siggraph computer graphics*, volume 21, 163–169. ACM.

Machado, P.; Correia, J.; and Romero, J. 2012. Expression-based evolution of faces. In *Evolutionary and Biologically Inspired Music, Sound, Art and Design*. Springer. 187–198.

Miller, G. A. 1995. Wordnet: a lexical database for english. *Communications of the ACM* 38(11):39–41.

Mordvintsev, A.; Olah, C.; and Tyka, M. 2015. Inceptionism: Going deeper into neural networks. *Google Research Blog. Retrieved June* 20.

Mouret, J.-B., and Clune, J. 2015. Illuminating search spaces by mapping elites. *arXiv preprint arXiv:1504.04909*.

Nguyen, A.; Yosinski, J.; and Clune, J. 2015a. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, 427–436. IEEE.

Nguyen, A.; Yosinski, J.; and Clune, J. 2015b. Innovation engines: Automated creativity and improved stochastic optimization via deep learning. In *Proceedings of the Genetic and Evolutionary Computation Conference*.

Pugh, J. K.; Soros, L.; Szerlip, P. A.; and Stanley, K. O. 2015. Confronting the challenge of quality diversity. In *Proc. of the Genetic and Evol. Comp. Conference*.

Razavian, A. S.; Azizpour, H.; Sullivan, J.; and Carlsson, S. 2014. Cnn features off-the-shelf: an astounding baseline for recognition. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on*, 512–519. IEEE.

Saunders, R., and Gero, J. S. 2001. The digital clockwork muse: A computational model of aesthetic evolution. In *Proceedings of the AISB*, volume 1, 12–21.

Secretan, J.; Beato, N.; D'Ambrosio, D. B.; Rodriguez, A.; Campbell, A.; and Stanley, K. O. 2008. Picbreeder: Collaborative interactive evolution of images. *Leonardo* 41(1):98–99.

Stanley, K. O. 2007. Compositional pattern producing networks: A novel abstraction of development. *Genetic programming and evolvable machines* 8(2):131–162.

Stiny, G., and Gips, J. 1971. Shape grammars and the generative specification of painting and sculpture. In *IFIP Congress (2)*, volume 2.

Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; and Rabinovich, A. 2015. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1–9.

Yosinski, J.; Clune, J.; Nguyen, A.; Fuchs, T.; and Lipson, H. 2015. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*.

Yumer, M. E.; Asente, P.; Mech, R.; and Kara, L. B. 2015. Procedural modeling using autoencoder networks. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, 109–118. ACM.

Zhou, B.; Lapedriza, A.; Xiao, J.; Torralba, A.; and Oliva, A. 2014. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, 487–495.