DeepCloud

Ardavan Bidgoli, Pedro Veloso, and Shenghui Jia

Problem

One of the most interesting topics in Machine Learning (ML) for design are generative models. In contrast to discriminative models, which only learns how to solve the learning task (classification, clustering, dimensionality reduction, etc.), generative models also learn how the data was generated, enabling the sampling of synthesized data based on the original distribution. Examples of generative models are Principal Component Analysis (PCA) , Autoencoders (AE), and Generative Adversarial Networks (GAN, Goodfellow et al., 2014), with all its variations and applications (Pix2Pix, CycleGAN, PGGAN, DAGAN, etc.)

However, the potential of generative models in design is still unexplored. Most of the advancements in generative design systems are problem-oriented (search, optimization, etc.) or rule-oriented (shape grammars, swarm models, cellular automata, etc.). There are almost no design application based on big data and few researches investigate design exploration with data-driven generative systems. As a consequence of this gap, there are no standards for modes of interaction, performance, and design representation with ML generative models for design.

Literature Review

While ML provides techniques that can be directly applied to the evaluation of building inhabitants data and post occupancy (Davis, 2016), recently, they have also been incorporated as a component of generative design systems. Sjoberg et al. (2017) use ML both for visualization and to support design optimization. Their workflow applies (1) a supervised neural network to predict the user selection of the population for a Genetic Algorithm, (2) a PCA to embeds the solutions in three-dimensional space and (3) Density-based spatial clustering to identify clusters with high-performance. Harding and Derix (2011) developed an algorithm to generate the layout of an exhibition hall. They use a SOM to embed the feature space of future exhibitions, converting it into a planar graph. They also apply a custom growing neural network to clusters the multiple graphs of the future exhibitions according to similar topologies.

However, these examples do not use ML techniques as the core of the generation process. One notable exception, is the work of Mohamed Zaghloul (2015), which uses a SOM to explicitly generate and organize new design alternatives of a villa. The input of the SOM neural-network is the box geometry of six design alternatives and the output layer is a two-dimensional grid with 15 by 15 cells containing the original input and generated non-linear morphing samples between.

DeepCloud

To contribute with generative models for design, we developed a general design tool that incorporates recent advancements in deep generative models to conceive 3d point-cloud objects in real-time: DeepCloud. After researching deep neural networks that can learn with point cloud data (Su et al., 2015; Maturana and Scherer, 2015, Yi et al., 2016; Qi et al., 2017a and Qi et al., 2017b), we opted to use the Autoencoder (AE) developed by Achlioptas et al. (2017). It combines a relatively shallow and simple architecture with custom layers and loss functions (Earth Mover's Distance and Chamfer Distance) to operate with point clouds.

In the interface, DeepCloud contains intuitive tools for the manipulation of high-dimensional data, aiming at the generation of suprising and meaningful design objects.

References

Achlioptas, Panos, Olga Diamanti, Ioannis Mitliagkas, and Leonidas Guibas. 2017. "Learning Representations and Generative Models for 3D Point Clouds." ArXiv Preprint ArXiv:1707.02392.

Davis, Daniel. 2016. "Evaluating Buildings with Computation and Machine Learning." In Posthuman Frontiers: Proceedings of the 36th ACADIA Conference. Ann Arbor.

Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. "Generative Adversarial Nets." In Advances in Neural Information Processing Systems, 2672–2680.

Harding, John, and Christian Derix. 2011. "Associative Spatial Networks in Architectural Design: Artificial Cognition of Space Using Neural Networks with Spectral Graph Theory." Design Computing and Cognition' 10, 305–323.

Maturana, Daniel, and Sebastian Scherer. 2015. "VoxNet: A 3D Convolutional Neural Network for Real-Time Object Recognition." In , 922–28. IEEE. https://doi.org/10.1109/IROS.2015.7353481.

Qi, Charles R., Hao Su, Kaichun Mo, and Leonidas J. Guibas. 2017a. "Pointnet: Deep Learning on Point Sets for 3d Classification and Segmentation." Proc. Computer Vision and Pattern Recognition (CVPR), IEEE 1 (2): 4.

Qi, Charles Ruizhongtai, Li Yi, Hao Su, and Leonidas J. Guibas. 2017b. "Pointnet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space." In Advances in Neural Information Processing Systems, 5105–5114.

Sjoberg, Christian, Christopher Beorkrem, and Jefferson Ellinger. 2017. "Emergent Syntax: Machine Learning for the Curation of Design Solution Space." In Disciplines Disruption: Proceedings of the 37rd ACADIA Conference, 552–61.

Su, Hang, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. 2015. "Multi-View Convolutional Neural Networks for 3d Shape Recognition." In Proceedings of the IEEE International Conference on Computer Vision, 945–953.

Yi, Li, Hao Su, Xingwen Guo, and Leonidas Guibas. 2017. "SyncSpecCNN: Synchronized Spectral CNN for 3D Shape Segmentation." In Computer Vision and Pattern Recognition (CVPR).

Zaghloul, Mohamed. 2015. "Machine-Learning Aided Architectural Design: Self-Organizing Map Generate in between Design Alternatives." In Proceedings of the 18th Generative Art Conference, 283–93. Venice. https://www.generativeart.com/ga2015_WEB/machine-learning_Zaghloul.pdf.

Process

The first step in DeepCloud is to setup and train the AE (fig.1). In our setting, the encoder should learn how to compress a point cloud of 2048 points into vectors of size 32 (latent space) and the decoder should be able to reconstruct the initial point cloud, using the Chamfer distance to evaluate it. In this R^{32} space, point clouds with similar characteristics are embedded in the same neighborhood, and each dimension is associated with certain characteristics. We trained this AE with point clouds generated from a database of models (shapeNET) and synthesized models, using categories such as chairs, hats and buildings.

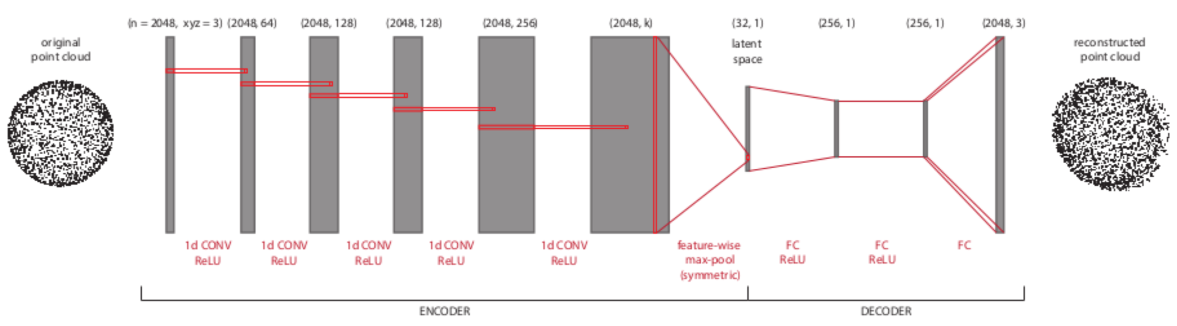


Fig.1: Architecture of the Autoencoder

The trained AE is stored in the back-end of the DeepCloud application, written in Python with Tensorflow. It can receive a vector in R^{32} and translate it to the respective point cloud (fig. 2). This is a generative model, so it can generate not only the original objects in the data base but also sample new synthesized objects.

In the front-end, DeepCLoud is a web interface (fig.2 and 3) that enables the user to manipulate the latent space representation and generate new point clouds. The users can (1) select existing objects from the data base as a starting point for a new model. Then, they can use two tools: (2) manipulate features, which enables the control of the features of the latent space, modifying specific aspects of the base model; (3) interpolation, which enables the combination of multiple models to generate a hybrid with shared characteristics. For future developments, a (4) GAN will be implemented to generate new starting points for the design.

The interface uses an analogic controller with sliders and knobs to provide an intuitive exploration of the latent space (fig. 2 and 3).

The chairs developed in DeepCloud can be saved in the database, which provides a new base models for feature manipulation and interpolation. Besides, they can be exported for 3d printing (fig.2, 5 and 6).

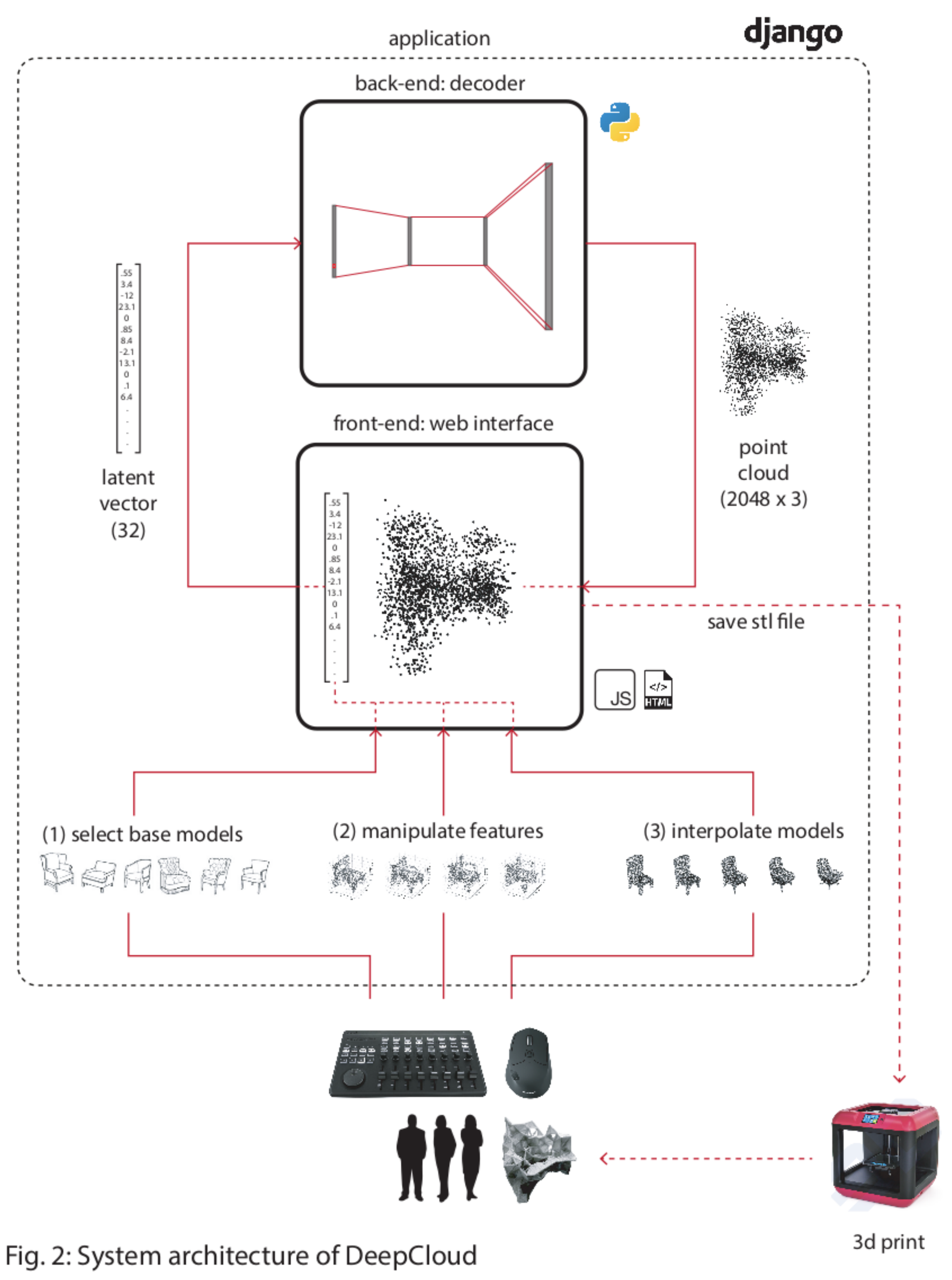


Fig. 2: System architecture of DeepCloud

Results

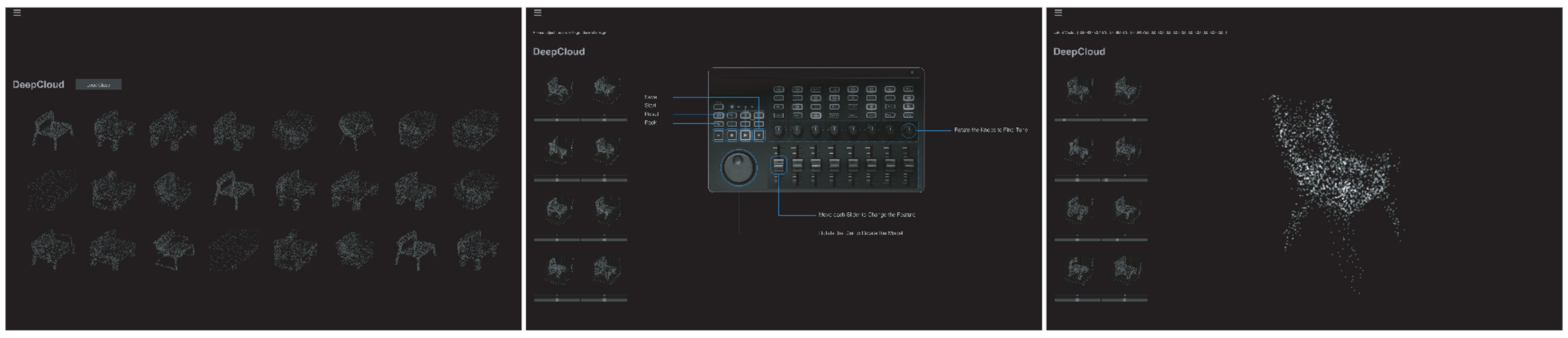
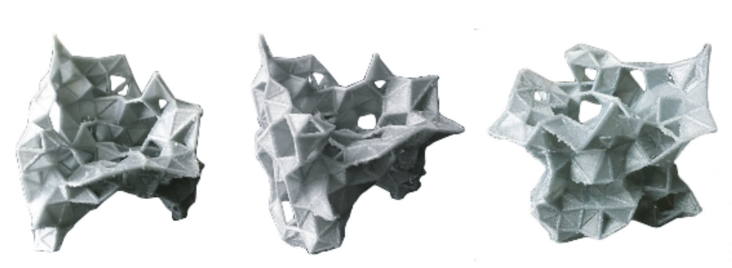


Fig. 3: Screenshots of the interface



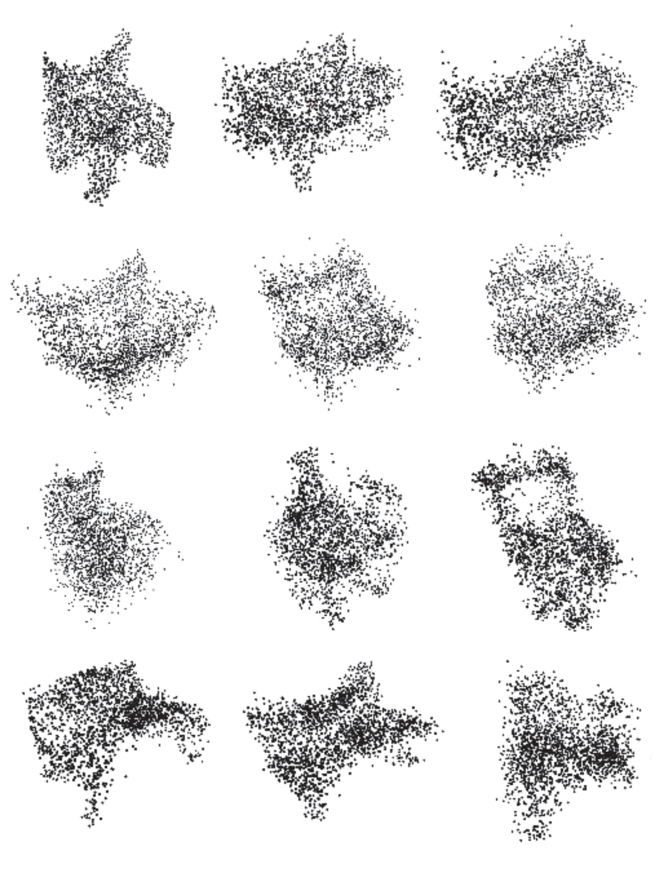


Fig. 4: Point clouds generated by DeepCloud

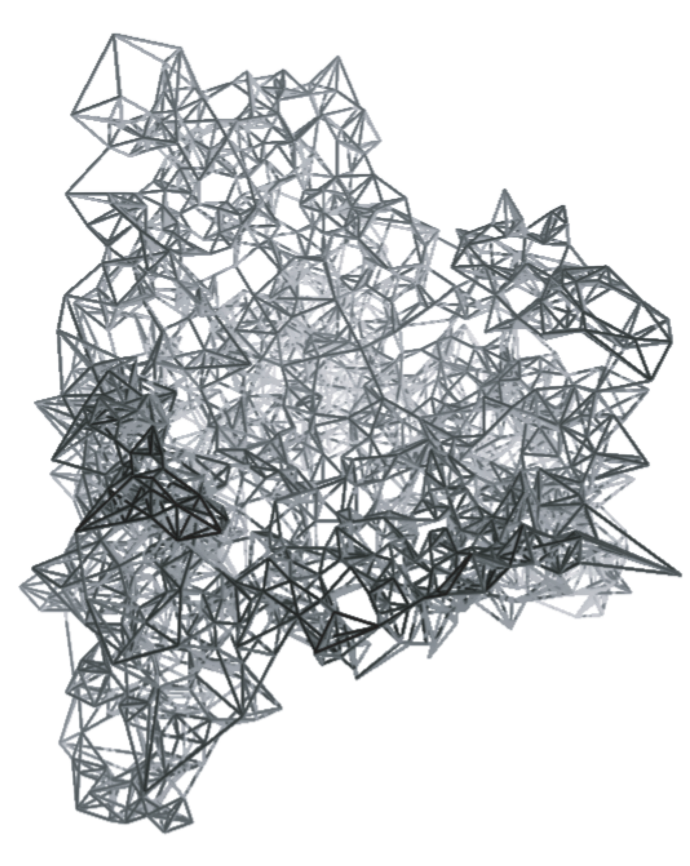


Fig. 5: Extracting structure from a point cloud

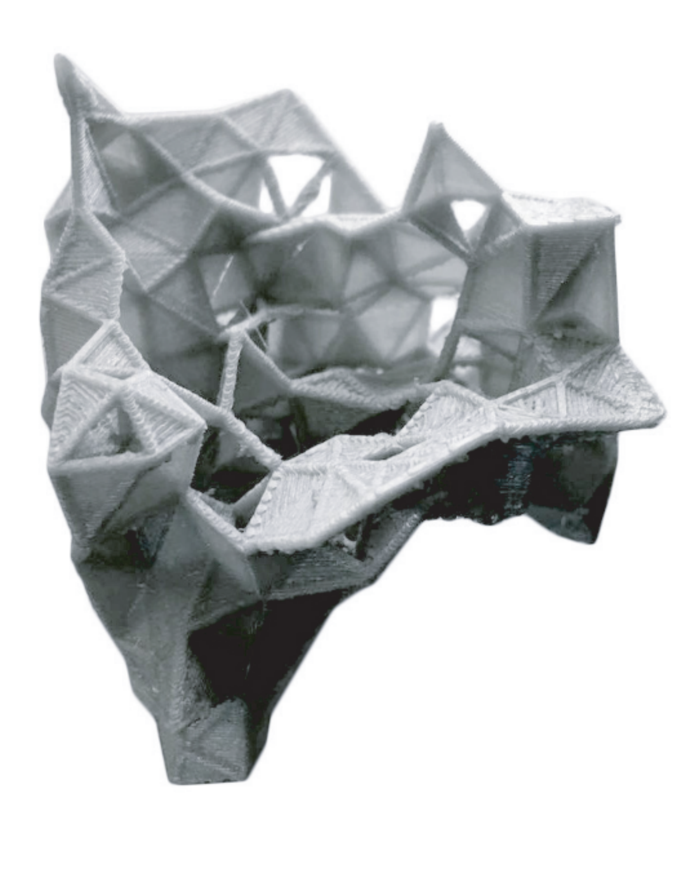


Fig. 6: 3d printing of the selected chair