FROM EXPLORATION TO INTERPRETATION

Adopting Deep Representation Learning Models to Latent Space Interpretation of Architectural Design Alternatives

JIELIN CHEN[1] and RUDI STOUFFS[2]
[1,2]*Department of Architecture, National University of Singapore*
[1]*chen.jielin@u.nus.edu* [2]*stouffs@nus.edu.sg*

Abstract. An informative interpretation of the hyper-dimensional design solution space can potentially enhance the cognitive capacity of designers with respect to both conventional design practice and the research domain of computational-aided generative design. However, the hitherto research of design space exploration has had limited focus on the interpretation of the hyper solution space per se due to the knowledge gap pertaining to representation and generation. Representation learning techniques, as a core paradigm in the statistically empowered domain of machine learning, possess the capability of extracting a convoluted probabilistic distribution of hyperspace with latent features from unorganized data sources in a generalized manner, which can be an intuitive modus operandi for a structural interpretation of the intricate latent design solution space and benefit the challenging task of architectural design exploration. We examine and demonstrate the potential capabilities of representation learning techniques for the interpretation of latent architectural design solution space with consideration of disentanglement and diversity.

Keywords. Design space exploration; latent space interpretation; representation learning; deep generative modelling; generative architectural design.

1. Introduction

The exploration of design alternatives is a routine practice for designers, the workflow of which can be perceived as an iterative framing process of design alternatives given explicit and implicit sets of contextual constraints, which are typically tricky to exhaustively codify. The nature of architectural design is discrepant from most problem-solving tasks in that the latent problems in architectural design are often ill-structured; defining the problem space is an intricate part of the design task (Eastman 2001). Hence, the process of design exploration can be depicted as mapping a series of latent design solution spaces and searching for optimal solutions within them. As the scope of manual design exploration is limited by the cognitive capacity of human designers, it grants chances for computational support and assistance (Woodbury & Burrow 2006).

PROJECTIONS, Proceedings of the 26th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA) 2021, Volume 1, 131-140.

Generative design has been developed as a powerful assistant for designers in the field of architecture and engineering. Two successive generations of generative design epistemology so far have leveraged the design exploration capability of generative systems. The first generation of generative design frameworks involves rule-based or performance-driven algorithms to transform a set of design objectives into a solid design product with certain criteria constraints. To satisfy such constraints, a series of objective functions need to be predefined with a set of design attributes (Sönmez 2015). In general, the first generation of generative design exploration regards design practice as a pathfinding task within a latent solution space at a local level using predefined navigation mechanisms, which has intrinsically hindered its potential generalization in practical applications. Such generative design systems can lead to a potential cognitive overload of its users, due to the large number of generated alternatives which need further clustering and grouping (Erhan et al. 2017).

In fact, without a global cognition of the hyper-dimensional latent design solution space, the scope of design exploration is doomed to be limited. Hence, it is necessary to advocate an epistemic shift and focus on the interpretation of the hyper solution space with a possibly global perspective. However, research in design space exploration has had limited focus on the interpretation of the design space per se ascribed to the knowledge gap pertaining to representation and generation (Woodbury & Burrow 2006). The emerging second generation of generative design schemes, benefitting from the boost of deep generative modelling in the field of machine learning, shows signs of being embedded with an intrinsic epistemology pertaining to the interpretation of the hyper solution space. The notion of deep generative modelling refers to a statistical modelling technique, which is capable of learning the latent feature distribution space from raw data in any format, and generating previously non-existing data samples from the estimated feature distribution space (Bengio et al. 2013; Gui et al. 2020).

Representation learning techniques, as a core machine learning paradigm playing a pivotal role in deep generative modelling, possess the capability of extracting convoluted probabilistic distributions of hyperspace with latent features from unorganized data in a generalized manner and map any feature vector from a random distribution to a semantically organized one (Bengio et al. 2013). This can be an intuitive modus operandi for structural interpretation of the intricate hyper design solution space and benefitting the challenging task of architectural design exploration. As deep representation learning can be used to leverage the abstraction of significant design features as decodable factors (Bengio et al. 2013; Han et al. 2019), the learned hyper-dimensional feature space can possibly represent the architectural design solution space with latent features reflecting any physically expressible design codes.

2. Research methodology

The rationale behind both deep generative modelling and representation learning is to learn a non-linear mapping between a latent feature distribution space and the real data space (Goodfellow et al. 2014; Xiao et al. 2019). A qualified deep representation learning model is required to automatically learn a disentangled

representation from the input data and minimize the distance between real and learned feature distribution (Tsai et al. 2019). A series of works have been done to address the automatic variation disentanglement task in the machine learning domain. Bau et al. (2020) have identified that some neural units of intermediate layers of a trained deep generative model are specialized to synthesize specific visual concepts. Meanwhile, Yang et al. (2020) found that a highly-structured semantic feature hierarchy can spontaneously emerge without any supervision from the learned representations of deep generative models adopting layer-wise stochasticity, such as BigGAN (Brock et al. 2019) and StyleGAN (Karras et al. 2019). The mechanism of layer-wise stochasticity indicates that the input latent codes for data synthesis are fed into all network layers of a deep neural net instead of only fed into the first layer, which enables better interpolation and disentanglement of the learned semantic features. Thus, we identify deep generative models with layer-wise stochasticity as our experimental backbones.

Meanwhile, both BigGAN and StyleGAN have the capability to synthesize diverse and high-quality image data. The difference between them is that the former is a conditional deep generative model while the latter is unconditional. BigGAN concatenates the latent code with a class-embedded code before feeding it to the generator (Brock et al. 2019). StyleGAN first maps a randomly sampled latent code from an initial latent space Z to an intermediate high dimensional space W using an auxiliary multilayer perceptron network before feeding it into the generator. The intermediate space W possesses a stronger disentanglement property than the initial space Z (Karras et al. 2019). Such property offers better interpolation and disentanglement of the learned latent semantic features, and further allows for scale-specific control of the data synthesis. The recently proposed StyleGAN2 model (Karras et al. 2020) further advances StyleGAN with better conditioning for the mapping from latent space to real data space.

As extensive coverage of the real data space ensures closer approximation to the right scope of data that a human architect would experience, access to large-scale datasets with quasi-exhaustive coverage and diversity of the real data space can be essential. Hence, we adopted a customized dataset of outdoor architectural imagery (roughly 37K images) retrieved from ArchDaily® and Dezeen® (online repositories of architectural projects worldwide) to train the selected models with desired properties, namely BigGAN, StyleGAN and StyleGAN2. We use image data as they are comparatively easier to acquire in bulk. All training was performed on Nvidia Tesla V100 16GB GPU or Nvidia Tesla V100 32GB GPU. StyleGAN2 outperforms the other two models evaluated using Frechet inception distances (Heusel et al. 2017; BigGAN: 175.5853; StyleGAN: 10.1867; StyleGAN2: 8.7180; lower is better). All subsequent experiments here described were performed with StyleGAN2 (Figure 1).

3. Interpretation of hyper-dimensional latent architectural design space learned by deep generative models

StyleGAN (and StyleGAN2) can automatically learn an unsupervised separation of high-level factors and stochastic variation in the data space which control the appearance of synthesized data (Karras et al. 2019). We leverage the ability of

StyleGAN2 to achieve semantic architectural design exploration by providing a rigorous interpretation of the semantic features emerging in the learned latent space of the well-trained deep generative model and manipulating the semantics in the learned latent space for architectural design attribute exploration.

Figure 1. Generated architectural image samples using well-trained StyleGAN2.

The generator of a trained deep generative model can be formulated as a deterministic function $g : Z \rightarrow X$, where $Z \subseteq R^d$ denotes the d-dimensional latent space and X the synthesized data space, with each synthesized sample x possessing certain sets of semantic information (Shen et al. 2020). A semantic scoring function $f_S : X \rightarrow S$ with $S \subseteq R^m$ representing the semantic latent space with m semantics can be introduced to link the latent space Z and the latent semantic space S with

$$s = f_S(g(z)), \tag{1}$$

where s denotes the semantic scores and z the sampled latent code. The linear interpolation between two latent codes z_1 and z_2 forms a linear direction in the latent space which can be regarded as the normal vector of a separation boundary, namely a hyperplane separating the corresponding semantic features. The distance between a sample latent code z and a hyperplane can be defined as

$$d(n, z) = n^T z, \tag{2}$$

where $n \in R^d$ is a unit normal vector and $d(\cdot, \cdot)$ can be both positive or negative; the semantic attribute will be reversed when z moves across the hyperplane. $f_S(g(z))$ and $d(n, z)$ are expected to be dependent upon each other, having

$$f_S(g(z)) = \lambda d(n, z), \tag{3}$$

where the scalar $\lambda > 0$ measures the unit magnitude of the latent semantic variation concomitant with the changing distance between the latent code z and the corresponding hyperplane. To disentangle multiple semantics, the latent semantic space needs to be further extended as

$$\mathbf{s} \equiv f_S(g(\mathbf{z})) = \mathbf{\Lambda} \mathbf{N}^T \mathbf{z}, \tag{4}$$

where $\mathbf{s} = [s_1, \ldots, s_m]^T$ indicates multiple semantic scores, and $\mathbf{N} = [n_1, \ldots, n_m]$ contains the unit normal vectors of the corresponding separation boundaries of semantic features, while $\mathbf{\Lambda} = diag(\lambda_1, \ldots, \lambda_m)$ denotes the diagonal matrix containing the coefficients between the variation of semantic score and corresponding distance to the boundary. Assuming that the distribution of latent code samples $\mathbf{z}$ can be approximated as Gaussian distribution $N(\mathbf{0}, \mathbf{I}_d)$, then $\mathbf{s} \approx N(\mathbf{0}, \mathbf{\Sigma s})$ is a multivariate normal distribution. Consequently, the manipulation of any semantic attribute of the synthesized data can be done by moving the latent code z along the unit normal vector of the hyperplane of the

corresponding semantic feature (Figure 2) with

$$\check{\mathbf{z}} = \mathbf{z} + \alpha\mathbf{n}. \tag{5}$$

To extract architectural semantic attributes from the learned latent space of the well-trained deep generative model, we further employ an auxiliary wide-resnet-based attribute prediction model trained with the SUN attribute database, which is manually labelled with 102 pre-defined scene attributes (Patterson et al. 2014; Zagoruyko & Komodakis 2017). The attribute prediction model was trained with multi-label losses to predict multiple attributes synchronously, and all attributes of the pre-trained prediction model were learned as bi-classification tasks. We use the adopted attribute prediction model to assign semantic attribute scores to the synthesized images and use the obtained attribute scores to further retrieve latent semantic boundaries inside the learned hyper architectural design space.

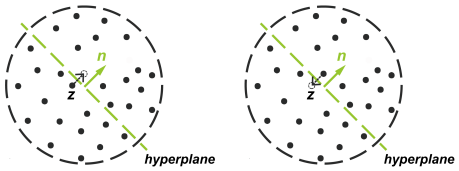

Figure 2. Manipulation of a semantic attribute of the synthesized data by moving the latent code z alongside the unit normal vector n of the hyperplane of the corresponding semantic feature. The synthesis will be either more positive ($\alpha > 0$; left) or negative ($\alpha < 0$; right).

Given the well-trained StyleGAN2 model for architectural imagery, we synthesize 300K images with randomly sampled 512-dimensional latent codes z, to pseudo exhaustively cover the learned distribution of the latent space. The aforementioned attribute predictor is used to calculate attribute scores for all synthesized images. For each attribute, the images are sorted by the corresponding scores. The 2,000 samples with highest scores are chosen as positive candidates and the 2,000 with lowest scores as negative samples, so as to avoid false prediction of ambiguous candidates as the pre-trained attribute predictor might not be entirely accurate. 70% of the selected candidates are randomly chosen as training set to train a linear support vector machine (SVM), bringing forth a binary classification decision boundary, namely the aforementioned separating hyperplane of each semantic attribute. The other 30% of the selected samples are used to verify the performance of the SVM. The inputs for the SVM training are the initial latent codes z for image synthesis, while the binary labels are acquired using the attribute predictor, indicating the presence of a target attribute in the corresponding synthesized image. The normal directions of all learned semantic attribute boundaries are normalized to unit normal vectors for further attribute manipulation experiments of the synthesized images.

3.1. DISENTANGLEMENT OF HIERARCHICAL SEMANTIC ATTRIBUTES

As we have previously discussed, deep generative models taking in layer-wise latent codes can spontaneously embed the semantics of the learned latent

space in a hierarchical manner from various abstract levels, which facilitates disentanglement of semantic features and manipulation of the synthesis process (Yang et al. 2020). To interpret the hierarchical semantics embedded in the learned generative representations for architectural image synthesis, 22 attributes relevant to architectural design are selected from the original 102 attributes of the SUN database (Patterson et al. 2014) and collected into five categories based on their characteristics (Table 1).

Table 1. Selected attributes relevant to architectural design categorized into five categories.

Characteristic	**Attributes**
Configuration	vertical_components, horizontal_components, symmetrical, cluttered_space
Openness	open_area, semi-enclosed_area, enclosed_area
Texture	glossy, matte, sterile, rugged_scene, rusty, moist, dry
Senses	soothing, stressful, warm, cold, man-made,natural
Material	brick, tiles, concrete, metal, wood, glass

We synthesized a series of images using the well-trained deep generative model with identical initial latent code z while moving z across a set of specified attribute boundaries (Figure 3). The outputs have verified that it is possible to controllably manipulate the synthesis process with specified attributes.

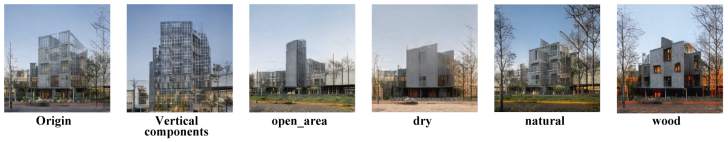

Figure 3. Synthesized samples of manipulating specified attributes with the same latent code z using the deep generative model trained on the architectural image dataset.

Figure 4 further illustrates some synthesized samples with distances close to and away from the learned attribute boundaries. For each row of samples, the middle one is the original synthesis with initial latent code z, while the left and right are synthesized results by moving z alongside the negative and positive direction of the unit normal vector n of the learned attribute boundary respectively. Each step indicates one unit distance away from the attribute boundary.

Figure 4. Synthesized samples manipulated at two specified attributes with distances close to and away from the learned attribute boundaries.

As previously mentioned, StyleGAN2 possesses a more disentangled latent space W based on the latent space Z of conventional deep generators which feeds

the latent code $w \in W$ to every convolutional layer with varied transformations; There are a total of 14 convolutional layers in the generator of the well-trained StyleGAN2. To measure the relevance level of each variation factor concerning every layer, attributes are re-scored per layer to quantify the correlation between the layer-wise representation and the corresponding semantic attribute. The normalized score shows that the layers of the generator are specialized to synthesize attributes hierarchically: Layers close to the input layer determine the configuration and openness of the architectural image, layers closer to the output layer control the attribute-level variations, and the material scheme is mostly rendered at layers closest to the output layer (Figure 5). This sequence is fairly consistent with a human architect's perception of the design process.

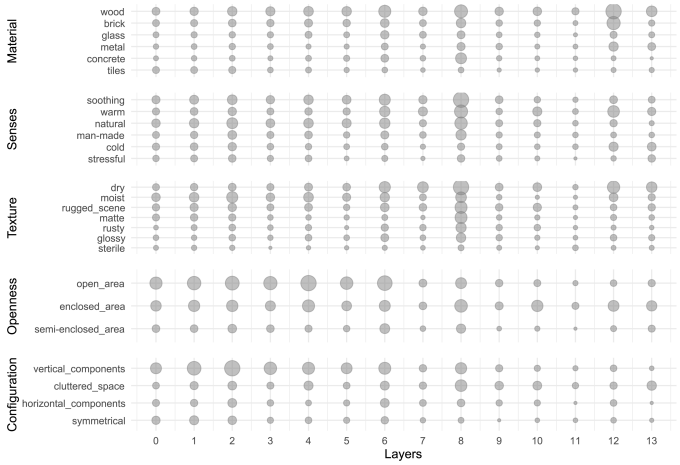

Figure 5. The correlation level between the layer-wise representation and the semantic attributes of synthesized images, the attributes in each characteristic group are sorted in descending order based on the overall level of relevance of the corresponding attribute.

3.2. DISENTANGLEMENT OF CORRELATED SEMANTIC ATTRIBUTES

As can be seen from the previous section, the manipulation results of some attributes are somehow intertwined with each other. It would be ideal to manipulate certain target attribute of the synthesis while keeping other attributes unchanged, which requires disentanglement of correlated semantic attributes. According to the previous discussion, different semantics *s* could be disentangled only when $\mathbf{\Sigma s}$ is a diagonal matrix and the unit normal vector of the separating hyperplanes of each semantic attribute $\{n_1, \ldots, n_m\}$ are orthogonal (linearly independent) with each other. Semantics violating this condition are correlated with each other and their entanglement level can be measured with $n_i^T n_j$.

As aforementioned, the latent space W of StyleGAN2 is not restricted to any predefined distribution and can learn a more disentangled representation, which enables layer-wise manipulation of the identified semantic features and can thus

provide controllable attribute editing. By layer-wise manipulating the latent code alongside certain semantic boundaries, we can visually examine how the synthesis varies at different layers. Figure 6 shows some synthesized samples with initial latent code z moving across specified semantic boundaries at the bottom, lower, upper, and top layers. These turn out to be mostly consistent with the conclusion drawn in section 3.1: 'vertical_components' can be significantly manipulated at bottom layers but have less impact at later levels of the model, whilst 'natural' can be best controlled at upper layers. Manipulating the latent code at less relevant layers can also lead to a change of image content. However, the change may be irrelevant to the expectation, f.i., manipulating the 'wood' attribute at any layers changes the synthesized image content, but the changes at middle layers relate to the building facade's configuration, the changes at upper layers relate to the texture of the facade, while the top layers contribute to the colour scheme. An interesting inference can be deduced from the observation: As the wood attribute can be relevant to both the texture and colour of the building facade, it might be possible to further decompose a singleton attribute feature based on its intrinsic definition.

To further quantify the correlation between different attributes, two metrics are adopted from Shen et al. (2020). First, the cosine similarity between two attribute normal vectors is defined as $\cos(n_1, n_2) = n_1^T n_2$, where n_1 and n_2 stand for the unit normal vectors. A value of zero indicates that the two unit normal vectors are orthogonal with each other, namely, they are fully disentangled. The cosine similarity matrix of the attribute boundaries of the learned latent feature space (Figure 7 left) can help to explain some of the observed phenomena, f.i., 'wood' attribute is relatively highly correlated with 'rugged_scene' with cosine similarity value 0.22, which can explain why manipulating the 'wood' attribute at middle layers can change the configuration of the building facade (Figure 6), considering that 'rugged_scene' has high correlation level at middle layers (Figure 5).

Figure 6. Synthesized samples with latent codes moving alongside the unit normal vector of two specified attribute boundaries at the bottom, lower, upper, and top layers of the well-trained model respectively.

Second, to examine the correlation of the synthesized attribute distribution, each attribute score is treated as a random variable, and the attribute scores obtained from all 300K synthesized images are used to calculate the Pearson correlation coefficient of every two semantic attributes respectively (Figure

7 right). We note the high correlation of ‘soothing’ and ‘natural’ attributes (correlation coefficient 0.79), ‘warm’ and ‘dry’ (0.79), and ‘glass’ and ‘glossy’ (0.77). The correlation among different attributes is similar for both metrics, which indicates that the adopted supervised approach for disentangling semantic attributes in the learned latent space is effective and accurate.

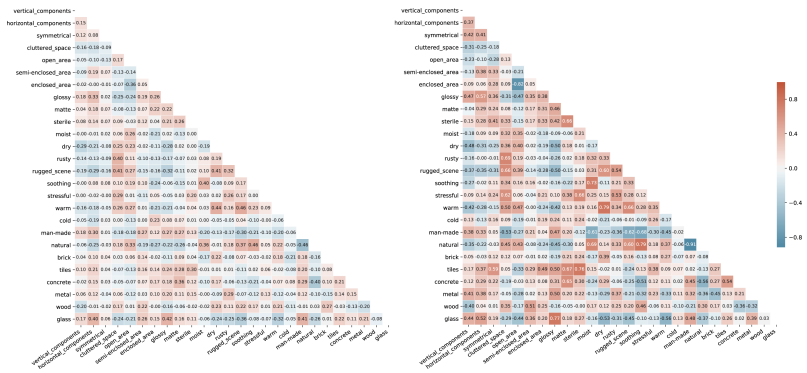

Figure 7. Cosine similarity matrix of the learned attribute boundaries (left) and correlation matrix of synthesized attribute distribution (right) for the well-trained deep generative model.

4. Discussion and summary

We have examined the potential capabilities of different representation learning-related techniques for latent architectural design space interpretation, and have demonstrated the multifaceted possibilities of leveraging representation learning to manipulate the exploration of architectural design alternatives controllably. This study’s long-term goals are to advocate a generic epistemic shift for more diverse and manipulatable design alternative generation and to promote interactive toolsets permitting user control over the synthesis process given that any physically expressible design codes can be intaken. This epistemic perspective might also be able to tackle the generalization issue of conventional generative design paradigms. Moreover, by coupling representation learning with optimization algorithms, it can be promising to achieve a more robust and efficient design space interpretation and exploration system.

Last but not least, the fact that the adopted attribute prediction model is not customized for architectural image recognition has hindered the accuracy and relevance of the attribute boundary retrieval task during the design space interpretation process. It would be worthwhile to develop an attribute predictor with an orientation of architectural design in the future. In addition, although it would be easy to expand the list of attributes for the predictor as assistance for latent design space interpretation, it is difficult to exhaustively explore the entire latent semantic space in a supervised manner. As such, it can be beneficial to introduce unsupervised approaches to disentangle the semantic features of latent architectural design solution space in future works.

Acknowledgements

The computational work for this article was partially performed on resources of the National Supercomputing Centre, Singapore (https://www.nscc.sg).

References

Bau, D., Zhu, J.Y., Strobelt, H., Zhou, B., Tenenbaum, J.B., Freeman, W.T. and Torralba, A.: 2020, "On the Units of GANs (Extended Abstract)" . Available from <https://arxiv.org/abs/1901.09887>.

Bengio, Y., Courville, A. and Vincent, P.: 2013, Representation learning: A review and new perspectives, *IEEE transactions on pattern analysis and machine intelligence*, **35**(8), 1798-1828.

Brock, A., Donahue, J. and Simonyan, K.: 2019, "Large Scale GAN Training for High Fidelity Natural Image Synthesis" . Available from <https://arxiv.org/abs/1809.11096>.

Eastman, C. and Computing, D. 2001, New directions in design cognition: studies of representation and recall, *in* C. Eastman, W. Newstetter and M. McCracken (eds.), *Design knowing and learning: Cognition in design education*, Elsevier, 147-198.

Erhan, H., Chan, J., Fung, G., Shireen, N. and Wang, I.: 2017, Understanding Cognitive Overload in Generative Design-An Epistemic Action Analysis, *Proceedings of CAADRIA 2017*.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y.: 2014, Generative adversarial nets, *Advances in neural information processing systems*, 2672-2680.

Gui, J., Sun, Z., Wen, Y., Tao, D. and Ye, J.: 2020, "A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications" . Available from <https://arxiv.org/abs/2001.06937>.

Han, Z., Shang, M., Liu, Y.S. and Zwicker, M.: 2019, "View Inter-Prediction GAN: Unsupervised representation learning for 3D shapes by learning global shape memories to support local view predictions" . Available from <https://arxiv.org/abs/1811.02744>.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. and Hochreiter, S.: 2017, Gans trained by a two time-scale update rule converge to a local nash equilibrium, *Advances in neural information processing systems*, 6626-6637.

Karras, T., Laine, S. and Aila, T.: 2019, A style-based generator architecture for generative adversarial networks, *Proceedings of CVPR 2019*, 4401-4410.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J. and Aila, T.: 2020, Analyzing and improving the image quality of stylegan, *Proceedings of CVPR 2020*, 8110-8119.

Patterson, G., Xu, C., Su, H. and Hays, J.: 2014, The sun attribute database: Beyond categories for deeper scene understanding, *International Journal of Computer Vision*, **108**(1-2), 59-81.

Shen, Y., Gu, J., Tang, X. and Zhou, B.: 2020, Interpreting the latent space of gans for semantic face editing, *Proceedings of CVPR 2020*, 9243-9252.

Sönmez, N.O.: 2015, Architectural layout evolution through similarity-based evaluation, *International Journal of Architectural Computing*, **13**(3-4), 271-297.

Tsai, Y.H.H., Liang, P.P., Zadeh, A., Morency, L.P. and Salakhutdinov, R.: 2019, "Learning Factorized Multimodal Representations" . Available from <https://arxiv.org/abs/1806.06176>.

Woodbury, R.F. and Burrow, A.L.: 2006, Whither design space?, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing: AI EDAM*, **20**(2), 63.

Xiao, Z., Yan, Q. and Amit, Y.: 2019, "Generative Latent Flow" . Available from <https://arxiv.org/abs/1905.10485>.

Yang, C., Shen, Y. and Zhou, B.: 2020, "Semantic Hierarchy Emerges in Deep Generative Representations for Scene Synthesis" . Available from <https://arxiv.org/abs/1911.09267>.

Zagoruyko, S. and Komodakis, N.: 2017, "Wide Residual Networks" . Available from <https://arxiv.org/abs/1605.07146>.