This is the author's copy. Full version: http://doi.org/10.1007/978-981-10-6190-5_22.
This paper won the Best Paper Award of CRIOCM 2016.

An optimization-based semantic building model generation method with a pilot case of a demolished construction

Xue, F.[1]*, Chen, K.[2], Liu, D.[3], Niu, Y.[4] and Lu, W.S.[5]*

Abstract: Emerging technologies like massive point cloud from laser scanning and 3D photogrammetry enabled new ways of generating 'as-built' building information models (BIM) for existing buildings. It is valuable but also challenging to generate semantic models from point cloud and images in automated ways. In this paper, we present a novel method called Optimization-based Model Generation (OMG) for automated semantic BIM generation. OMG starts from a semantic BIM component dataset and a target measurement such as point cloud, photographs, or floor plans. A fitness function is defined to measure the matching level between an arbitrary BIM model and the target measurement without object recognition. Combinations of digital components are then extensively generated as building models regarding semantic constraints. The fittest model that matches the target measurement best is the result of OMG. The proposed method was demonstrated in reconstructing a 3D model of a demolished building. Advantages of OMG include high-level automation, low requirement on measurement, relationship discovery for components, reusable component libraries, and scalability to new environments.

Keywords: Building information modeling (BIM); Automated building model generation; Derivative-free optimization; Black-box optimization; 3D model projection; Image structural similarity (SSIM) index; Demolished building; SketchUp API.

[1]* Xue, F.
Corresponding author, KB714, iLab, Faculty of Architecture, The University of Hong Kong, Pok Fu Lam, Hong Kong SAR
E-mail: xuef@hku.hk
Tel: (852) 2219 4174

[2] Chen, K.
Department of Real Estate and Construction, The University of Hong Kong, Pok Fu Lam, Hong Kong SAR

[3] Liu, D.
Department of Real Estate and Construction, The University of Hong Kong, Pok Fu Lam, Hong Kong SAR

[4] Niu, Y.
Department of Real Estate and Construction, The University of Hong Kong, Pok Fu Lam, Hong Kong SAR

[5]* Lu, W.S.
Corresponding author, Department of Real Estate and Construction, The University of Hong Kong, Pok Fu Lam, Hong Kong SAR
E-mail: wilsonlu@hku.hk
Tel: (852) 2859 7981

1 Introduction

Building information modeling (BIM) involves physical as well as functional characteristics of constructions and their components. BIM has been developed to facilitate construction management along the whole life-cycle[1] through as-required, as-designed, as-planned, as-built (or "as-is"), as-altered, and as-demolished BIM models. The as-built BIM model can provide critical building information about construction quality assessment, construction automation, energy consumption, green gas emissions, facility management, retrofitting planning, and renovation recommendation[2-5]. However, BIM was very recently adopted widely in architectural, engineering and construction (AEC) industrial practice. For example, the adoption of BIM expanded from 17% in 2007 to over 70% in 2012 in North America[6]. As a result, there is a large gap between the need and the availability of as-built BIM for many existing constructions. Thanks to the new technologies such as laser scanning, 3D photogrammetry and videogrammetry, the as-built BIM model now can be generated from point cloud and multiple images[7]. These new methods are much easier and more vivid than manual reconstruction. However, most of the available techniques that generate 3D models from point cloud and photogrammetry do not offer exploitable topological and semantic contents for BIM models[8].

In this paper, we present an Optimization-based Model Generation (OMG) method with an orientation of organizing semantic BIM components to fulfill the task of automated BIM generation. The remainder of this paper is organized with the following sections, related works, a general framework of the OMG method, a pilot case of a demolished building, and conclusions.

2 Related Works

A typical scan-to-BIM process is a semi-automated method which reply on both software (such as IMAGINiT "Scan to BIM" and CapturingReality) and BIM professionals to pre-process (register, merge/stitch, clean, decimate) data, to recognize (semantic labels), and to create/adjust BIM models[7,9]. Researchers also presented automated means in two categories in general to facilitate the reconstruction process. One category is "data-driven". An early example was that Baltsavias et al. represented[10] 3D blocks and roofs of buildings of University of Melbourne campus out from 2D IKONOS® satellite images. Most of automation functions of the available software on the market also belong to the "data-driven" category. The other category is "model-driven", such as a series of typical operations of pose adjustment by principle component analysis (PCA), silhouette extraction and merger, size adjustment, and position matching[11]. OMG is a model-driven method, yet with an optimization-based model generation mechanism.

Semantic BIM information, such as construction geometry and architectural design, has always been, implicitly or explicitly, included in the literature[7,9]. The mainstream methodology of existed research on semantic BIM was based on object recognition such as identifying component geometry or labeling semantic properties from input images or point cloud first[7-12]. Heuristic rules from human experts and automated knowledge discovery tools, such as support vector machines[12] and deep machine learning neural networks[13], were generally involved in BIM object recognition. However, the supporting technologies of object recognition are still in development. The data requirement is usually high and the performance varies case by case. For instance, Perez-Perez et al. reported[12] the best precision was between 79% to 92%, along with 2% to 31% labeled as "unrecognized", in automated labeling point cloud segmentations on average.

OMG, in contrast, does not involve object recognition process. In the framework of OMG, the BIM model is an entirety, and the target measure is another one. The objective of OMG is to pursue the best matching level between the two entireties as a systematic way of generating models. Off-the-peg BIM components and their semantic information are compiled for attaching, modifying, or removing objects during model generation in OMG.

3 OMG: A Semantic-Oriented Model Generation Framework

3.1 The general framework of OMG

In general, OMG requires two inputs, i.e. a target measure such as matching a point cloud or image set and a set of (usually excessive) semantic BIM component libraries, as shown in Figure 1. Typical semantic BIM data include geometric properties, surface pattern, architecture style, typical functions, typical supporting structures, material and thermodynamic properties, mechanical properties. The final output of OMG is a semantic BIM model. The first phase in OMG is the transformation of the target measure to a fitness function. Then a solver or algorithm, denoted as the dotted box in Figure 1, will try to optimize the fitness function in the second phase. The last phase is output the fittest model with necessary inverse transformation.

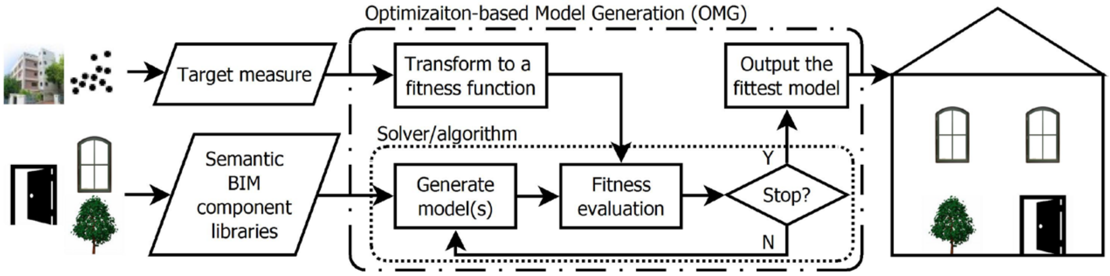

Figure 1. A general framework of optimization-based model generation (OMG).

In OMG, the task of building model generation is equivalent to the determination a set of parameters for each component. The parameter set may include a non-negative number of instances, geometric information (position, scaling, and rotation) of each instance, and connection/joint and topological relations to other component instances. The model generation task can hence be rewritten to

$$\text{optimize } f(x), \quad x = (x_1, \ldots, x_n) \in \Re^n,$$

where x denotes all parameters of all components in this research, n is the number of parameters. The fitness function f is defined as the matching level between an arbitrary model and the target measure. Hard constraints can also be introduced from semantic requirements and domain knowledge, such as the available technologies at the construction time, consistency to the architectural style, and instability of the structure. Hence, the task of building model generation can be extended to:

$$\begin{aligned} \text{optimize} \quad & f(x), \\ \text{subject to} \quad & C(x) \leq 0, \end{aligned}$$

where C means a function set representing the hard constraints.

3.2 Typical fitness functions and solvers

One of the most well-known functions f is the mean square error (MSE) which is defined as

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{Y}_i - Y_i\right)^2$$

where Y is the vector of target values (e.g., coordinates of a laser point) and $\hat{Y}=g(x)$ is the vector of model values (e.g., coordinates of the nearest point on the model surface). For instance, the MSE can measure Euclidean distance and difference of color between a model and a point cloud (or images). Competitive alternatives can also be found in some domains. For example, in the similarity test of 2D images, Wang et al.[14] presented the structural similarity (SSIM) index which integrated structure, luminance, and contrast measurements of 2D images. SSIM outperformed MSE in a few international image processing competitions.

The fitness function f is usually very sophisticated (derivative-free) and sometimes “opaque”. Therefore, many powerful solvers, such as IBM CPLEX™ and Gurobi Optimization™, cannot be directly adopted in OMG. In the literature of derivative-free optimization[15] (also known as black-box optimization, automatic parameter tuning, and model selection) and several international competitions [6], many well-known computational algorithms such as genetic algorithm (GA) were found with acceptable performances. Furthermore, surrogate methods, which adjust estimations of variables consistently, could find much more satisfied results, especially for the “expensive” functions which cost much effort (time and cost) to calculate. One typical surrogate method is the covariance matrix adaptation with evolution strategy (CMA-ES) proposed by Hansen and Ostermeier[16]. Solvers with surrogate methods such as CMA-ES are therefore recommended, whereas universal algorithms like GA can also be good candidates for “cheap” functions.

4 A Pilot Case

4.1 The target measure, component libraries and supporting software

The School of Tropical Medicine and School of Pathology, The University of Hong Kong (HKU) used to occupy a baroque-style two-storey architecture, as shown in Figure 2. The building was built on the main campus of HKU in 1919, later refurbished with an additional floor, and demolished in campus development decades ago. In this pilot test, this building is employed to describe the process of OMG.

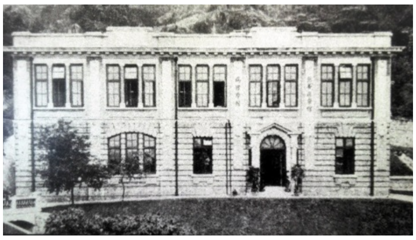

Figure 2. A demolished building of The School of Tropical Medicine and School of Pathology, The University of Hong Kong (640×360 pixels; Source: Exit A of MTR HKU railway station, photographed by an Android mobile phone in July 2016)

The task was set to reconstruct the BIM model of the front side of the building due to limited data from the target 2D image. More specifically, apparent components with width and height greater than 1 meter, including trees, front facade, the door and the windows should be included in the 3D model. Groups of components, as listed in Table 1, were collected from shared models of 3D Warehouse of SketchUp with a keyword filter “baroque”. Some semantic attributes such as type, material, surface glued to, typical size and typical locations were manually added to SketchUp dictionaries of components.

Table 1. A list of components extracted from shared models in SkechUp™ 3D Warehouse for the pilot case

Library name	Component name	Original model from 3D Warehouse (Contributor ID)	Attached semantic labels
Door	Door portico	Door (3) Classical Ottoman Osmanli Colonial (Mohamed EL Shahed); Blenheim Orangery and Function Rooms (Richard)	Typical size; Glued to: wall
Tree	Oak tree	Downy Oak (KangaroOz 3D)	Location: ground; Glued to: open space
	Palm tree	Royal Palm Tree (Yoshi Productions)	Location: ground; Glued to: open space
Wall	With h-sliding	Salm Palace (3dolomouc)	Typical size; Location: ground
	Smooth surface	Salm Palace (3dolomouc)	Location: 1/F and above; Glued to: wall
Window	Three-section	French Window (Architect)	Typical size; Glued to: wall
	Traditional	Mahogany Framed Window (Ben)	Typical size; Glued to: wall

The pilot test was implemented on SketchUp API (application programming interface, version 2016 PRO Ruby). The components were automatically adjusted with Ruby (version 2.0.0p648) scripts in SketchUp and then projected to 2D images (in 640×360 resolution). The similarities between projected images and the target image were measured by the SSIM function (Wang 2004). Scientific packages of OpenCV (Open source computer vision library, version 2.4.13) and

[6] See Ruhr-University Bochum, Germany: Black Box Optimization Competition. http://bbcomp.ini.rub.de/; See also Missouri University of Science and Technology, USA: Combinatorial Black-Box Optimization Competition. http://web.mst.edu/~tauritzd/CBBOC/

ruby-opencv (version 0.0.17) were employed to facilitate the measurement. A C++ implementation[7] of CMA-ES and its Ruby wrapper were adapted as the solver in OMG. Google Earth™ (version 7.1.5.1557) was used to present the resulting model. Stanford Protégé (version 5.0) was used to represent the semantic links in the resulting model.

4.2 Transformation and the solving procedure

Seven components, as shown in Table 1, were gathered for model generation. The numbers of component instances were set to 1 door portico, 1 tree, 2 walls, and 9 windows to simplify the pilot task. For each component instance, there were six real parameters, i.e., position and scaling on the three axes. The rotation parameters were omitted in this case due to the front face of the target building. According to estimation about the place where the photo was taken, the camera of SketchUp was located to (0, 850, 157.5) heading to (0, 0, 157.5) (unit in inch). The ground was defined as the horizontal plane z=0. The semantic relations of location and glued to were required as hard constraints. So the fitness function f is the dissimilarity between a projected image and Figure 2 to:

$$\begin{aligned} &\text{minimize} \quad && f = dissimilarity = 1 - SSIM \\ &\text{subject to} \quad && \text{Semantic constraints of location, size and glued-to} \end{aligned}$$

The model was generated increasingly by attaching one component each time, where ground components preceded facade instances. 200 trials of parameters were allowed for the CMA-ES solver in attaching a new component instance concerning f. The incremental generation procedure stopped when no new instance could lead to a reduction of f. The component instances were then fine-tuned by CMA-ES for an additional 2,000 trials before termination.

4.3 The pilot run and the result model

The pilot run was conducted on an Intel® i5-6500 CPU (3.20 GHz) computer. The CMA-ES solver spent 1 hour 3 minutes and 42.4 seconds (3,822.4s) to optimize f in 4,800 trials in total, as shown in Figure 3. After 400 trials from the beginning, the G/F wall and the door portico were attached to the model. Then a palm tree was attached yet was quickly replaced by an oak tree, because the attachment of the palm tree contributed an f = 0.8603 but the new oak tree conflicted with it in location and had a better f = 0.8392. The incremental generation phase stopped at the 2,800th trial. After 2,000 trials of the fine-tuning phase, the final model achieved a fitness f = 0.7772 by slightly changes of the door portico and a few windows automatically. The result of OMG was saved as a SketchUp file in 2.06MB.

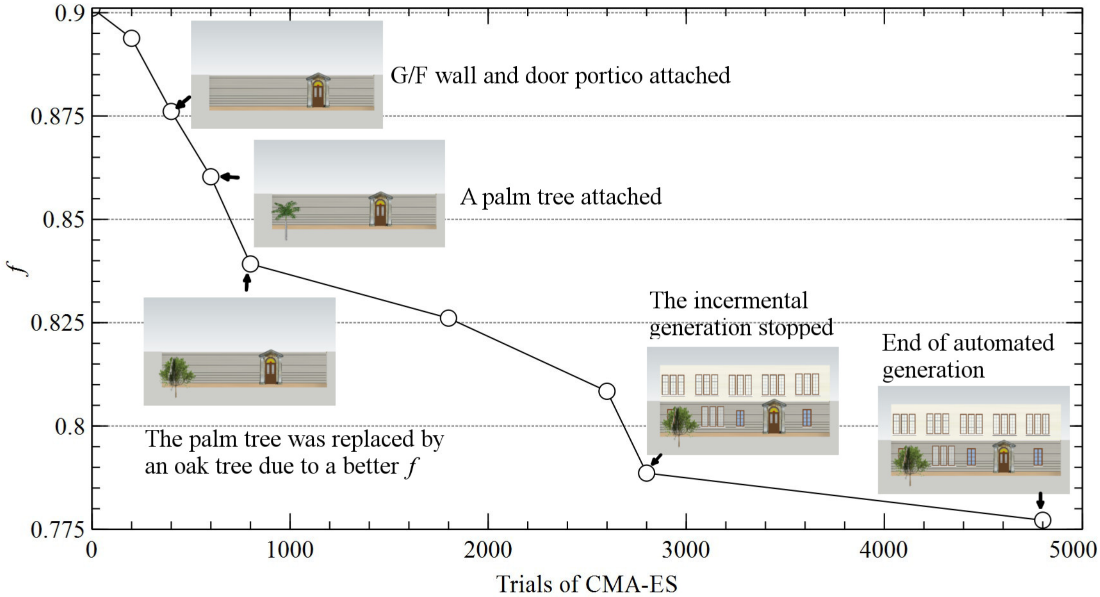

Figure 3. The automated optimization process of OMG with annotated SketchUp™ models in the pilot run (Incremental generation phase: 1~2,800; fine-tuning phase: 2,801~4,800)

[7] CMA-ESpp, see: https://github.com/AlexanderFabisch/CMA-ESpp

Some components in the resulting model, such as a three-section window glued to the G/F wall shown in Figure 4 (a), did not satisfy axial rules of baroque architecture. So the authors made a few manual amendments to the auto-generated front face of the building. The model of the whole building, as shown in Figure 4 (b), was estimated based on a survey map of HKU (1969~1976) and manually completed with the replicas of the auto-generated facade. An illustration of the building model, as shown in Figure 4 (c), described its historical location near today's Exit A2 of the HKU station and the Kadoorie Biological Science Building, HKU.

(a) Auto-generated front face in SketchUp (b) Manually completed building model (c) Illustrated on Google Earth

Figure 4. The resulting model of OMG

The semantic relations between the components were inherited during the model generation. The relations could also be exported as queryable ontology format such as the Web Ontology Language (OWL). The Ontology software like Protégé can then provide search and reasoning functions for the components and their relations in the generated model illustrated in Figure 4 (a). Figure 5 shows an example of the search result of term "Traditional". The "Traditional" window component and all the three instances are shown with "is-a" connections in Figure 5. The relations of "location" and "glued to" are presented as dashed links between component classes.

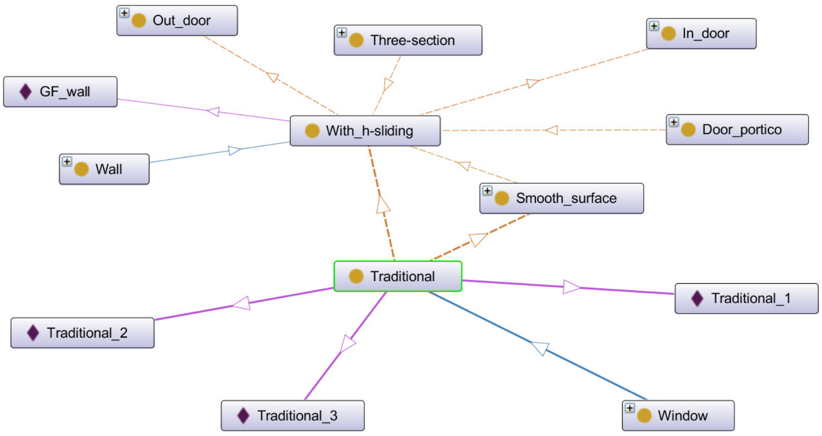

Figure 5. The semantic links in the auto-generated model illustrated in Stanford Protégé (Circle denotes a component class and a diamond stands for an instance/object)

4.4 Discussions

This case was a preliminary test of the feasibility of OMG. So some minor details such as decorative cornice and front door plants were excluded in this pilot test. The whole process of OMG used a couple of hours more than the approximate 1 hour spent by CMA-ES solver in this pilot. It was because the preparation of component libraries and post-optimization procedures brought extra time cost into the OMG process. Object recognition of the target image was not involved. However, the components were placed in the correct locations mostly and hence implicitly identified the objects with semantic labels and relations. For example, a window which was difficult to recognize visually was placed accurately behind the tree in the model. One limitation of this pilot was that the numbers of component instances were set to constants to simplify the task, although they can be a part of parameters to be determined according to the general OMG framework.

In comparison to typical scan-to-BIM process, OMG has several advantages. First, the automation level is elevated to a higher level. A saving of cost could also be expected from equipment, data gathering, and manpower. In OMG, BIM professionals need to be involved only in the early stages such as pre-process, component selection and definition, and definitions of fitness functions and constraints. Furthermore, the data requirement of OMG is also much more relaxed than that of scan-to-BIM (usually tens of millions of pixels). It is because many details can be inherited from components if defined. Other features of OMG include semantic data to meet BIM requirements, low requirement on measurement, relationship discovery for components, reusable component libraries, and scalability to new environments.

Disadvantages of OMG, nevertheless, can also be found. The first was the accuracy and availability of component libraries. Thanks to the antique building modelers over the world, most of the major components in this pilot case were extracted from shared models directly. Yet those components, such as the trees, did not perfectly match the target measure in fact. Another one was the quality of the model. The finite trials of computer program certainly cannot compare with BIM professionals' knowledge, experience, and insights in model generation.

5 Conclusion

This study presented a new semantic-oriented as-built BIM generation method named optimization-based model generation (OMG). In OMG, building model generation is regarded as an assembly of components from semantic BIM component libraries. The assembly of components is considered as a derivative-free function to fit target point cloud or images with respect to semantic constraints. Some computerized optimization algorithms can automatically find the best arrangements of component parameters and hence the corresponding 3D models. The result of OMG is a BIM model with geometric, topological and semantic data. A pilot case of a demolished building at HKU campus validated the main process of OMG.

This study is expected to enrich the research of automated BIM generation with an alternative framework to object recognition in point cloud and images. The findings of the pilot case were preliminary. Further research is needed to validate the transformation of the fitness function for other targets like point cloud, other advanced mathematical programming algorithms, and the effects of using semantic data as soft constraints.

Acknowledgement

The authors would like to express their gratitude to Mr. Pak Kei Hon at Estates Office, The University of Hong Kong for supporting documents of the pilot building.

References

[1] Lu, W., Fung, A., Peng, Y., Liang, C., and Rowlinson, S. (2014). Cost-benefit analysis of Building Information Modeling implementation in building projects through demystification of time-effort distribution curves. Building and Environment, 82: 317-327.

[2] Woo, J., Wilsmann, J., and Kang, D. (2010). Use of as-built building information modeling. Construction Research Congress 2010, 538-547. Banff, Alberta, Canada: ASCE.

[3] Lu, W., Peng, Y., Shen, Q., and Li, H. (2012). Generic model for measuring benefits of BIM as a learning tool in construction tasks. Journal of Construction Engineering and Management, 139(2): 195-203.

[4] Niu, Y., Lu, W., Chen, K., Huang, G. G., and Anumba, C. (2015). Smart construction objects. Journal of Computing in Civil Engineering, ASCE: 04015070.

[5] Kalyan, T. S., Zadeh, P. A., Staub-French, S., and Froese, T. M. (2016). Construction quality

assessment using 3D as-built models generated with project tango. Procedia Engineering, 145: 1416-1423.

[6] Bernstein, H. M., Jones, S. A., Russo, M.A., Laquidara-Carr, D., Taylor, W., Ramos, J., Healy, M., Lorenz, A., Fujishima, H., Fitch, E., and Buckley, B. (2012). The business value of BIM in North America: Multi-year trend analysis and user ratings (2007–2012). Bedford, MA, USA: McGraw-Hill.

[7] Volk, R., Stengel, J., and Schultmann, F. (2014). Building Information Modeling (BIM) for existing buildings—Literature review and future needs. Automation in Construction, 38: 109-127.

[8] Gimenez, L., Robert, S., Suard, F., and Zreik, K. (2016). Automatic reconstruction of 3D building models from scanned 2D floor plans. Automation in Construction, 63: 48-56.

[9] Barazzetti, L. (2016). Parametric as-built model generation of complex shapes from point clouds. Advanced Engineering Informatics, 30 (3): 298-311.

[10] Baltsavias, E., Pateraki, M., and Zhang, L. (2001). Radiometric and geometric evaluation of IKONOS GEO images and their use for 3D building modelling. Proceedings of Joint ISPRS Workshop on High Resolution Mapping from Space 2001. Hannover, Germany, 19-21 September (on CD-ROM)

[11] Liu, T. Zhao, D., and Pan, M. (2016). An approach to 3D model fusion in GIS systems and its application in a future ECDIS. Computers & Geosciences, 89: 12-20.

[12] Perez-Perez, Y., Golparvar-Fard, M., and El-Rayes, K. (2016). Semantic and geometric labeling for enhanced 3D point cloud segmentation. Construction Research Congress 2016, 2542-2552. San Juan, Puerto Rico: ASCE.

[13] Patraucean, V. (2016). Deep machine learning: Key to the future of BIM? Cambridge: University of Cambridge, Mar 14 2016. Accessed July 15 2016. http://www-smartinfra structure.eng.cam.ac.uk/news/viorica-patraucean-featured-in-infrastructure-intelligence-.

[14] Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing, 13 (4): 600-612.

[15] Conn, A. R., Scheinberg, K., and Vicente, L. N. (2009). Introduction to derivative-free optimization (Vol. 8). SIAM.

[16] Hansen, N., and Ostermeier, A. (2001). Completely derandomized self-adaptation in evolution strategies. Evolutionary Computation, 9 (2): 159-195.