ARCHITECTURE
+ ML

OHLHOFF RABAN - 000457528

Master in architecture 1st year

Digital Fabrication Studio

ULB-Faculté d'Architecture La Cambre-Horta
Place Eugène Flagey 19,
1050 Bruxelles

TABLE OF CONTENTS

STUDIO

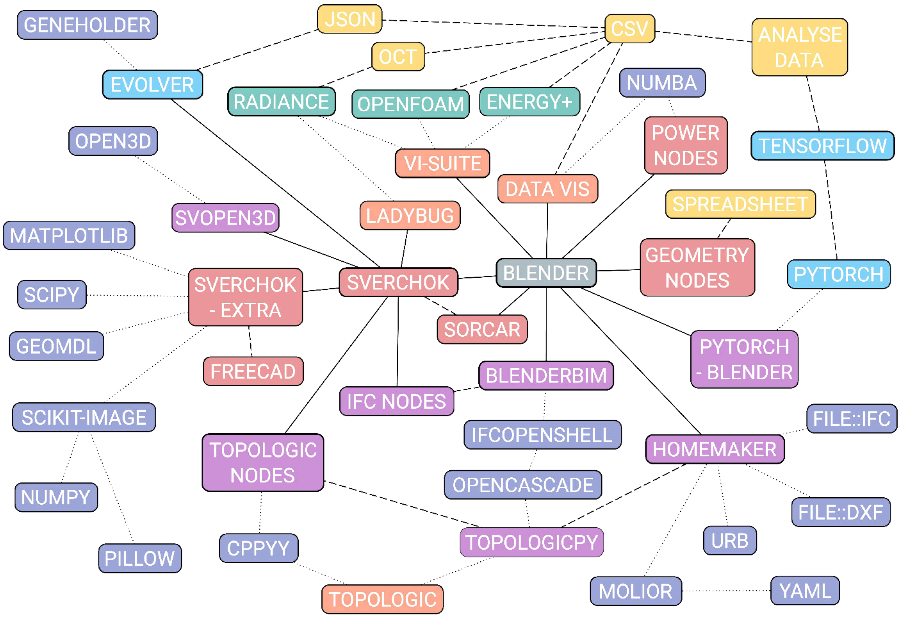
This book aims to explore the increasingly important interaction between machine learning and architecture. The following project was developed in the framework of the Digital fabrication studio at the Université libre de Bruxelles, which is characterized by an opening towards new technologies. This Studio poses interdisciplinary questions between technology, research and architecture that can be developed freely, without being constrained to remain in limited fields. The aim is to develop hypotheses about the future of architecture by taking into account all the fields that may be interesting to link in view of the project design process. In each aspect of this process, it is important to take into account the analysis of new scientific knowledge and to conduct in-depth research on the achievements of current science. Nevertheless, it is still important to base the experimentation on the inventions that have emerged throughout history.

The main issues of this workshop are not to invent new objects but rather to deepen the study of a specific topic with all existing publications and research in order to formulate a relevant research hypothesis. The aim is then to begin a process of experimentation that will be accompanied by detailed documentation. The aim of this process will be to provide answers to the questions posed beforehand. In this way, the experimentation cycle will advance the research and may eventually lead to an objective solution, but will in any case raise new questions that can be pursued later.

The workshop's operation is strongly linked to the Fablab of the Faculty of Architecture and thus gives access to many tools and means of digital design and experimentation. This place will also function as a library of knowledge shared by each individual and will therefore promote the genesis of collective knowledge. Due to the restructuring around the Covid-19 epidemy, the workshop is being transposed into virtual space. This requires a revision of the experimentation process.

PROCESS

First of all, the subject of artificial intelligence needs to be dealt with in depth. In today's world, AI is used as a kind of selling point, a kind of solution to all complex problems. Is this assessment true, or is this term simply attached to ideals of solutions?
Once these questions have been answered and a concrete concept has emerged from the abstract term, it became possible to think about beneficial links between architecture and machine learning that simplify existing design processes, drafts, simulations or constructive procedures.



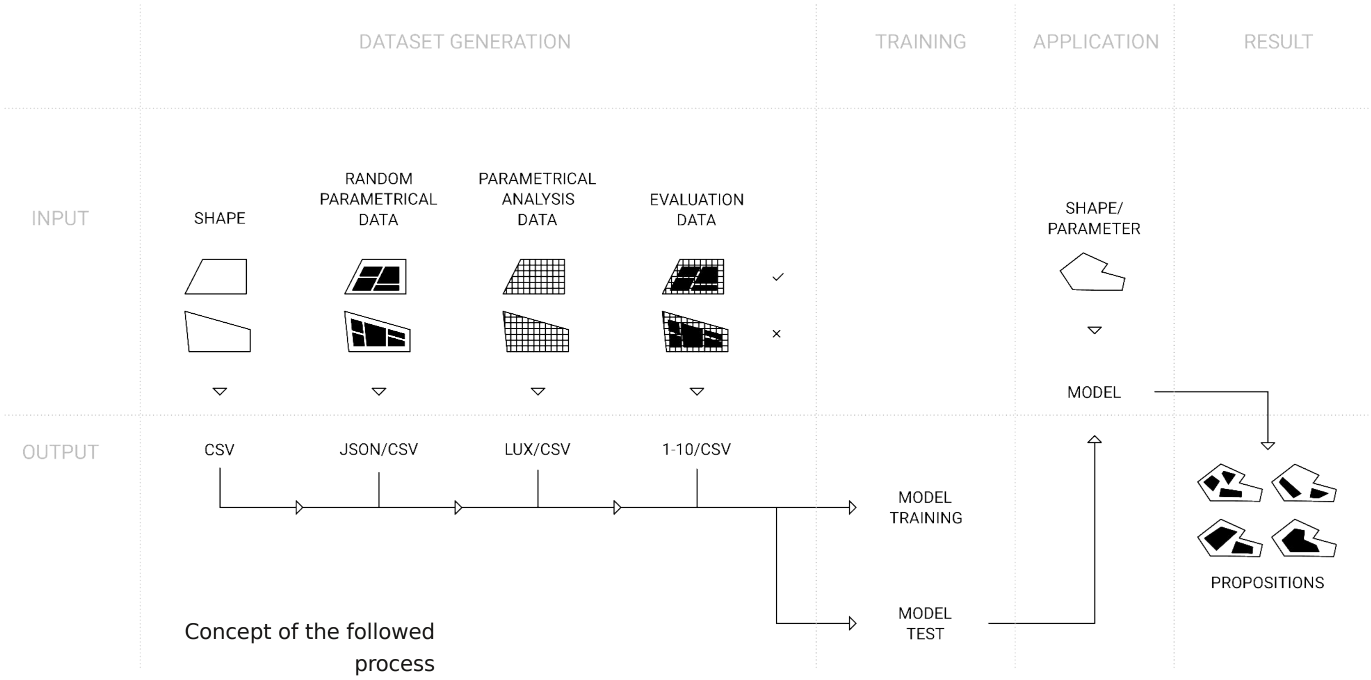
Schematic representation of software interactions

As important as a clear understanding of the issues, it is important to understand the available libraries and their functions in order to create a network diagram of the interactions. In this work we will work exclusively with open-source packages to allow the greatest possible transparency and to avoid the black box principle. Furthermore, the open-source community allows a proper and direct exchange with any contributor and a full understanding of the functions and operations. With the help of various forums and exchanges with developers, it was possible to gain a sufficient understanding of the software in question in a relatively short time and thus to advance the ideas of the project.

ABSTRACT

This project attempts to simplify the architectural design process by means of a prediction model and thus to carry out a pre-selection of certain volumetrics on the basis of desired characteristics. To achieve this, five steps had to be taken, each with different software support. First the parametric generation of a block of houses using Blender and Sverchok, an environmental simulation with Blender, Radiance, Energyplus and Vi-Suite, the data processing of the generated CSV files with pandas, the training and testing of a prediction model with SciKit learn, numpy, jupyter notebook and finally the reimposition of the result in Blender using Sverchok, TopologicPY and topologize. Special attention was paid to the uniform use of exclusively python-based libraries to ensure the best possible compatibility of the individual results. During the execution of the individual stages, any problems, suggestions, solutions, proposals for

improvement and, equally important, ideas for further exploration of the subject matter were noted, which are described in detail in the following book.
The results of the individual stages and ultimately the final result yielded several insights. On the one hand, Blender proved to be very capable as a central monitoring program thanks to its Python-based compatibility and accessible API with any libraries. It also showed that export, import, edition and transport of any data could be set up without any problems due to an open file format. The final result turned out to be satisfactory, but showed clear weaknesses in accuracy. However, these difficulties do not call the research into question and can be solved by logical optimizations. Optimization does not require starting from scratch, thanks to the compatibility of each operation, but can offer adaptation to more complex challenges by replacing or adding certain operations, libraries or values. Finally, it is clear that the process followed in this project, through its flexibility and functionality, provides an interesting approach in the union between architectural design and mechanical learning.



Concept of the followed process

HYPOTHESIS

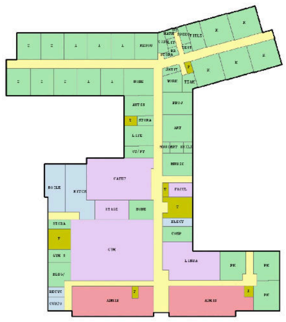
Using artificial intelligence to optimize traditional processes has become the norm in this day and age. Machine learning is no longer limited to computer science. Well-known application areas include mass production optimization, various applications in medicine, the military and, to a large extent, research.

It is therefore not surprising that Big Data-based optimization has also found its way into architecture. However, the application of AI in architecture is very versatile and can be useful in every project development process. For example, intelligent parameterization can be used to help find the right shape even before a concrete project is created. A mechanical analysis of the existing and framework conditions can be useful to determine an approximate volumetry. If the next step is the compartmentalization of the interior surface, it is possible to propose several optimal plans with intelligent logarithms and thus optimize thedesign. Furthermore, the analysis is not limited to the two-dimensional space and can therefore provide suggestions for optimal circulation or optimization of the incidence of daylight on all floors.
After the conceptual phase, it is also possible to optimize the IFC model with different algorithms. All these processes are no longer visions of the future but have become the norm, but often automated and therefore not visible to everyone.
After the conceptual phase, there are also many different applications of artificial intelligence. There are approaches that help construction workers to build or realize a smart home or city project via the Internet of Things.
In this work, I will mainly focus on the application of computer algorithms in the conceptual phase. Moreover, I consider that a specialization in generative design is interesting at this stage. However, I am concerned about this approach because my knowledge of computer science is far from adequate. Nevertheless, I think that a field trip on the subject could be useful and that I can learn some non-preconceived knowledge.
In the conceptual phase, there are also various applications of computer intelligence. First of all, the search for formulas can be facilitated by a database-based analysis of different conditions: Traffic, solar radiation, shadow formation, wind flow, heat formation, soil conditions, air quality or general aesthetic characteristics can be taken into account.

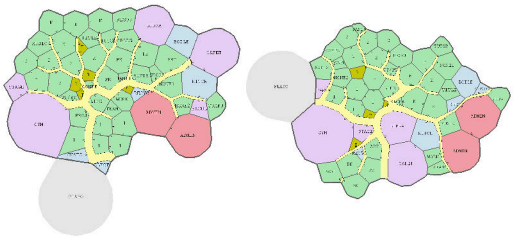
The main hypothesis of this work is to question if and to what extent machine learning algorithms can simplify, speed up and/or optimize the architectural design process. Is the increasing digitalization of architecture purposeful or will traditional values be lost? Is the application of AI again only an idealized label or are there concrete advantages on the side of the clients as well as on the side of the architect? Furthermore, this thesis aims to explore to what extent it is possible to generate a complete workflow from generation via simulation, analysis, prediction back to generation without resorting to proprietary software and thus to describe a step towards the democratization of architecture and its digital tools.

STATE OF THE ART

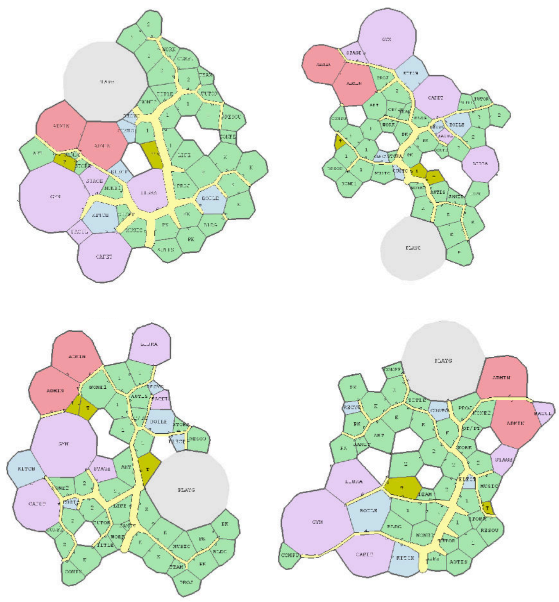
The company Finch3d has developed an algorithm that creates adaptive plans. Thus, it is enough to specify the number of people who will live in the apartment and the shape of the project. The algorithm then proposes plans for the interior. This is not an intelligent algorithm, which means that it is not a learning process, but rather rules linked to conditions.
In addition, Stanislas Chaillou has done extensive work on generating 2D plans using Artificial intelligence at the Harvard University in 2019. It is based on a principle of visual plan recognition by Weixin Huang and Hao Zheng in 2018. What is interesting about Chaillou's work is that the algorithm generates several different proposals and also learns through a feedback loop. It has also been experimenting with different architectural styles.
The idea of generative design has been explored by several protagonists. For example, Joel Simon optimized a school building based on two different criteria. The objective was to optimize the incidence of light and, in another experiment, to shorten the circulation. The results of the individual optimization researches were amazingly organic building forms and a compositional reshaping of the entire building.
For a project of a bridge in Amsterdam, Mx3d used a generative algorithm which, through static analysis and organic shapes, resulted in a number of optimally adapted bridge designs.
In the paper "Learning From Main Streets" Oh, Smith and Koile explain how machine learning can be used in urbanism.



original disposition



circulation optimization



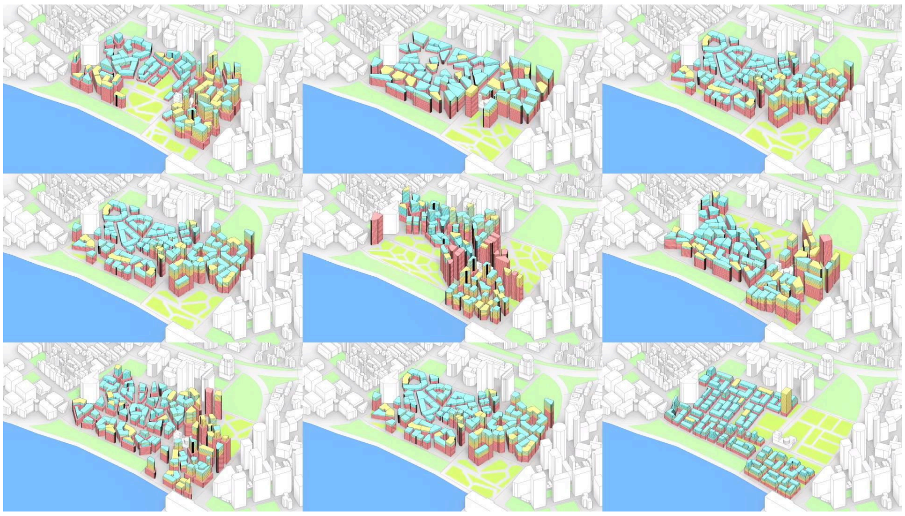
window optimization

The model is trained to analyze a complex urban network and its infrastructure on a large scale, enabling the architect to integrate his work into the environment in a context-sensitive way. In this example we are dealing with GIS based training data.
In the article 'Towards Machine Learning for Architectural Manufacturing in the Age of Industrie
4.0' published in the International Journal of Architectural Computing, Thomsen et al. describe potential implications of machine learning algorithms in the AEC industry. It is argued that there are two moments in the design process that can be optimized by intelligent mechanisms: the data creation and the fabrication process. Also interesting is the importance outlined in this article of evaluation functioning as a feedback loop to improve results.
To understand the term generative design in more detail, the paper 'Design Optimization in Early Project Stages.
A Generative Design Approach to Project Development' by Rohrmann. In summary, generative design processes are based on generative algorithms which belong to the family of evolutionary algorithms. The basis of these is simple and in principle not very different from the biological evolution we know: A population is generated which has different random characteristics and a defined mutation rate. In the following stage, the survival and fitness of the individuals is observed and evaluated. Finally, the number of fit genes is increased and the number of less fit genes is decreased.
The best contemporary example of a generative design approach in urban planning was developed by Sidewalk labs, a subsidiary of google. The goal of this project is similar to the approach of this work. Through parametric generation of bigdata based on defined characteristics such as open space, daylight and density, the architect is able to optimize in advance the further formal research and thus base any further design process on appropriate fundamentals. The architect involved in the design process is offered a variety of urban scenes, which in turn can be parameterized to meet the desired characteristics both aesthetically and functionally.
Finally, it is important to consider the available simulation and analysis software that meets the requirements: flexibility, compatibility and functionality. Radiance is a light analysis suite originally developed by Greg Ward in 1985 which has increasingly become the standard in architectural light simulation and the basis of many analysis tools, for example the Grasshopper plugin Ladybug and Openstudio. Advantages of this software are the open access to the source code and the open source license. Furthermore, Radiance's open file format .rad allows unlimited compatibility.

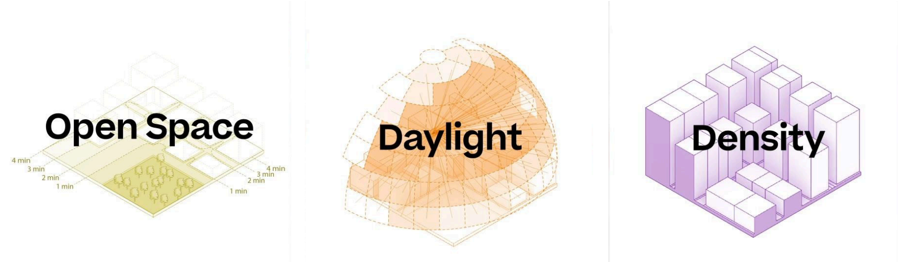
EnergyPlus is an open-source software specialized in energy management developed by the US department of energy. It provides analysis for heating, cooling, ventilation and water use in buildings. This is primarily a building internal simulation. Furthermore, Openfoam can be used for simulation of complex fluid flows (wind, water and heat), CalculiX for finite element analysis of constructive elements, Code Aster for multiphysical analysis (seismic analysis, acoustics, fatigue, stochastic dynamics, etc.) and JuPedSim a framework to investigate pedestrian dynamics.

CONCLUSION

The projects mentioned in the state of the art all provide very interesting and diverse approaches to the unification of artificial intelligence and architecture. Due to the limited nature of this work, the focus will be on the generative design approach of Sidewalklabs. However, it is not possible to reproduce the underlying processes, so this work will attempt to achieve similar results while disclosing each individual process. A focus is put on the adaptability of the process, in other words, what kind of simulation is chosen as a characteristic has no influence on the flow of the process, so no significant changes need to be made after adapting the objectives.



Different parametrically generated proposals from *Sidewalk Labs*

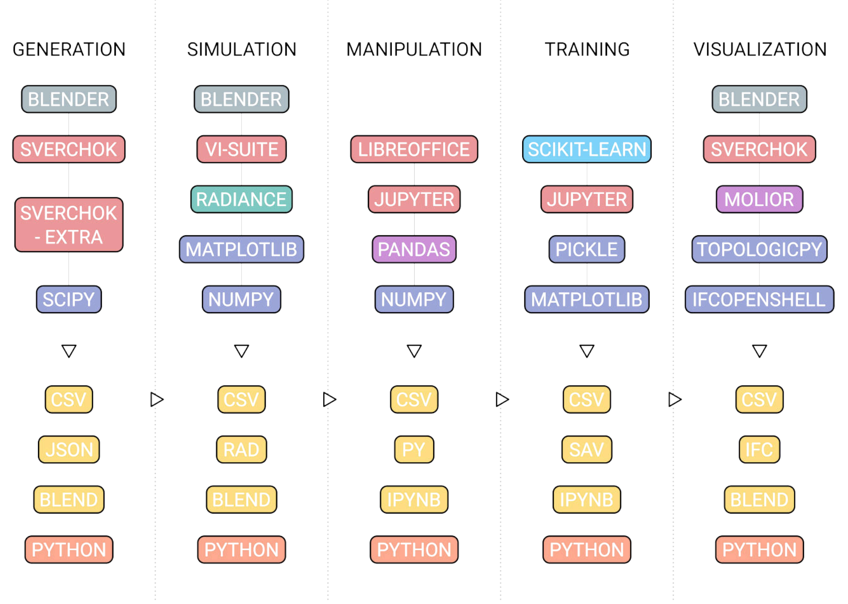


Main parameters for identifying suitable proposals from *Sidewalk Labs*

After a closer look at the existing examples of different applications of machine learning in the AEC industry, it becomes clear that fears of a loss of aesthetic qualities and a reduction of the architectural profession to a computer scientist are not confirmed. On the contrary, it even allows the participants to invest the additional energy available in the aesthetic research process and thus to produce work of higher quality. This means that the application of artificial intelligence has earned its place in architecture, engineering and construction and will definitely result in some interesting symbioses in the future.

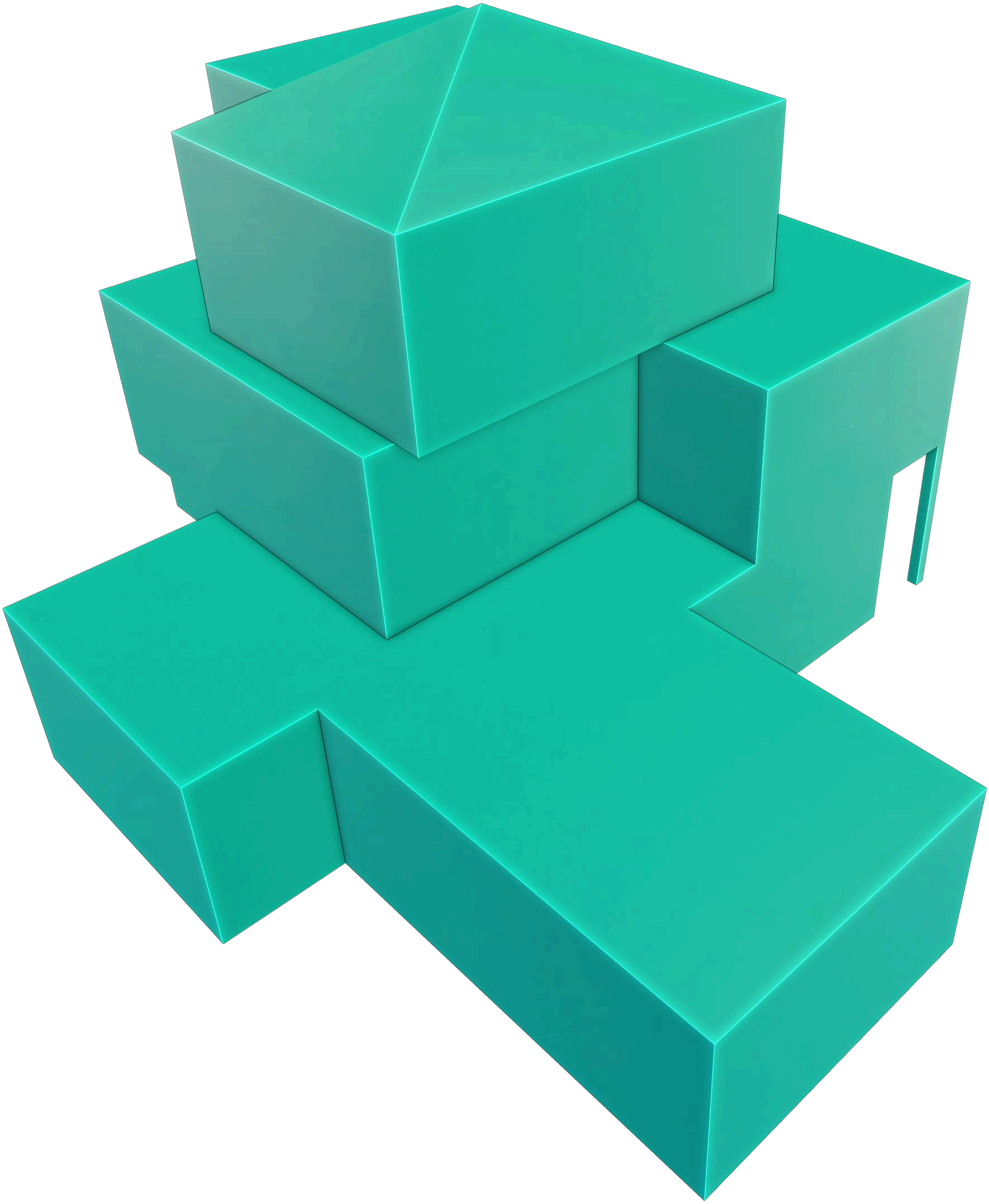
APPROACH

This work is an experimental approach, which means that it is not primarily about developing an optimized process, but rather about a critical analysis of the individual stages. Thus, by repeatedly questioning the progress, it will be possible to draw insights that will be helpful in the following stages. Furthermore, an essential focus is on answering the questions formulated in the hypothesis, which means that the individual steps must be reflected on several levels in order to see not only technical insights but also moral, ethical and social ones.
There is no previous knowledge in data science, coding, parametric generation, simulation, analysis and visualization, so this work is also a means to experiment to what extent it is possible for outsiders to adopt foreign tools in a limited time frame and what influence a focus on open-source software and compatabilty through open file formats has in the workflow shown.



OBJECTIVES

The experimental freedom explained above also results in flexibility with regard to the objectives. In general, there are objectives for each stage, but it does not indicate a failed step if they turn out differently than expected or formulated in advance. Thus, the parametric generation stage has as its hoped-for output the generation of randomly generated models that can be fixed by certain parameters. These can be the number of buildings, the number of inhabitants, the area, the volume or the shape of the building complex. The end result of this stage should be a Comma Separated Value (CSV) table that describes each situation as accurately as possible. In the next stage, the simulation will produce a new value that can be added to this table. In the fourth stage a model is trained as a result which describes the coherence of the individual values as accurately as possible by a graph. Finally, the last stage should allow an accurate visualization of the predicted scenario.

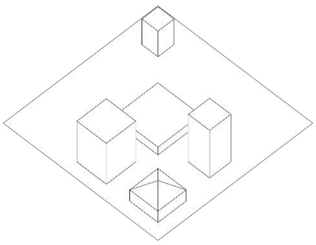
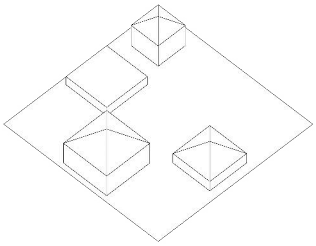
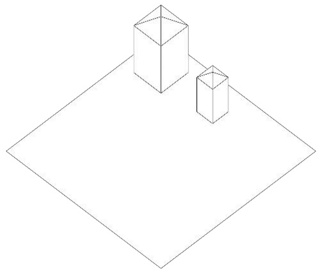
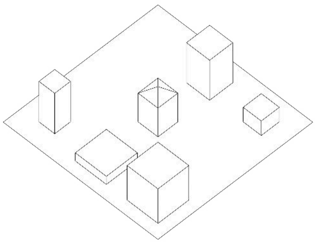
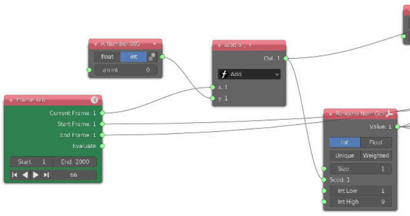
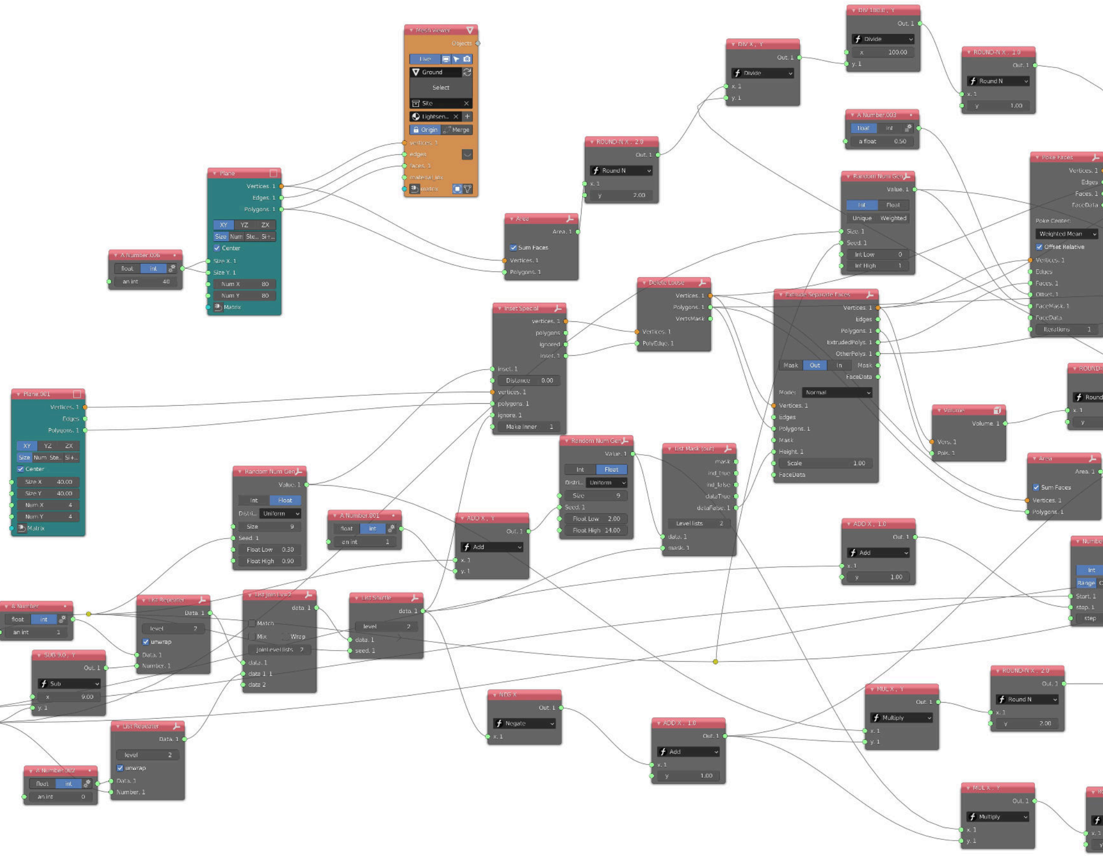
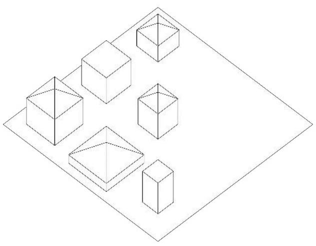
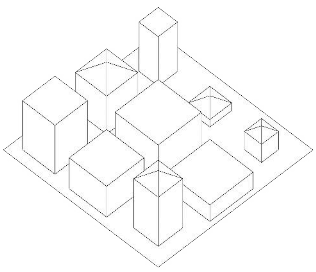
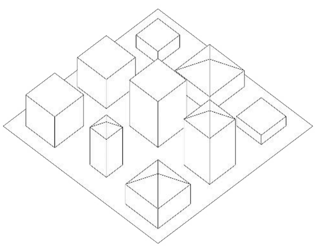
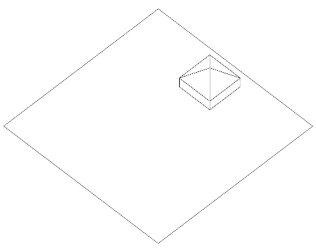
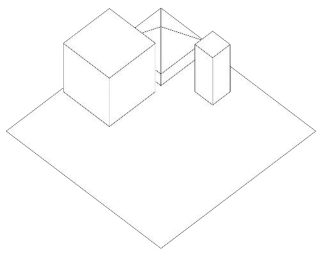


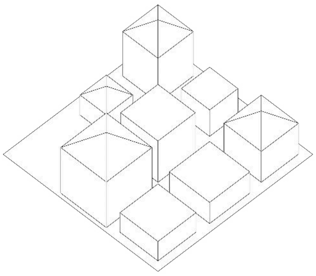
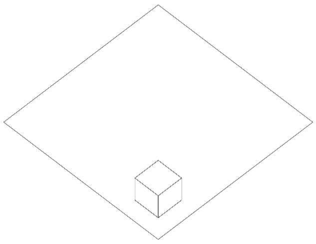
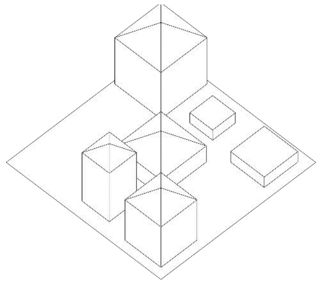
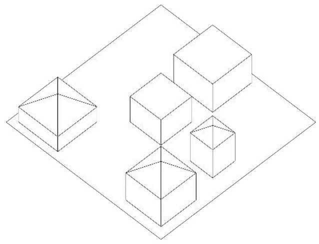
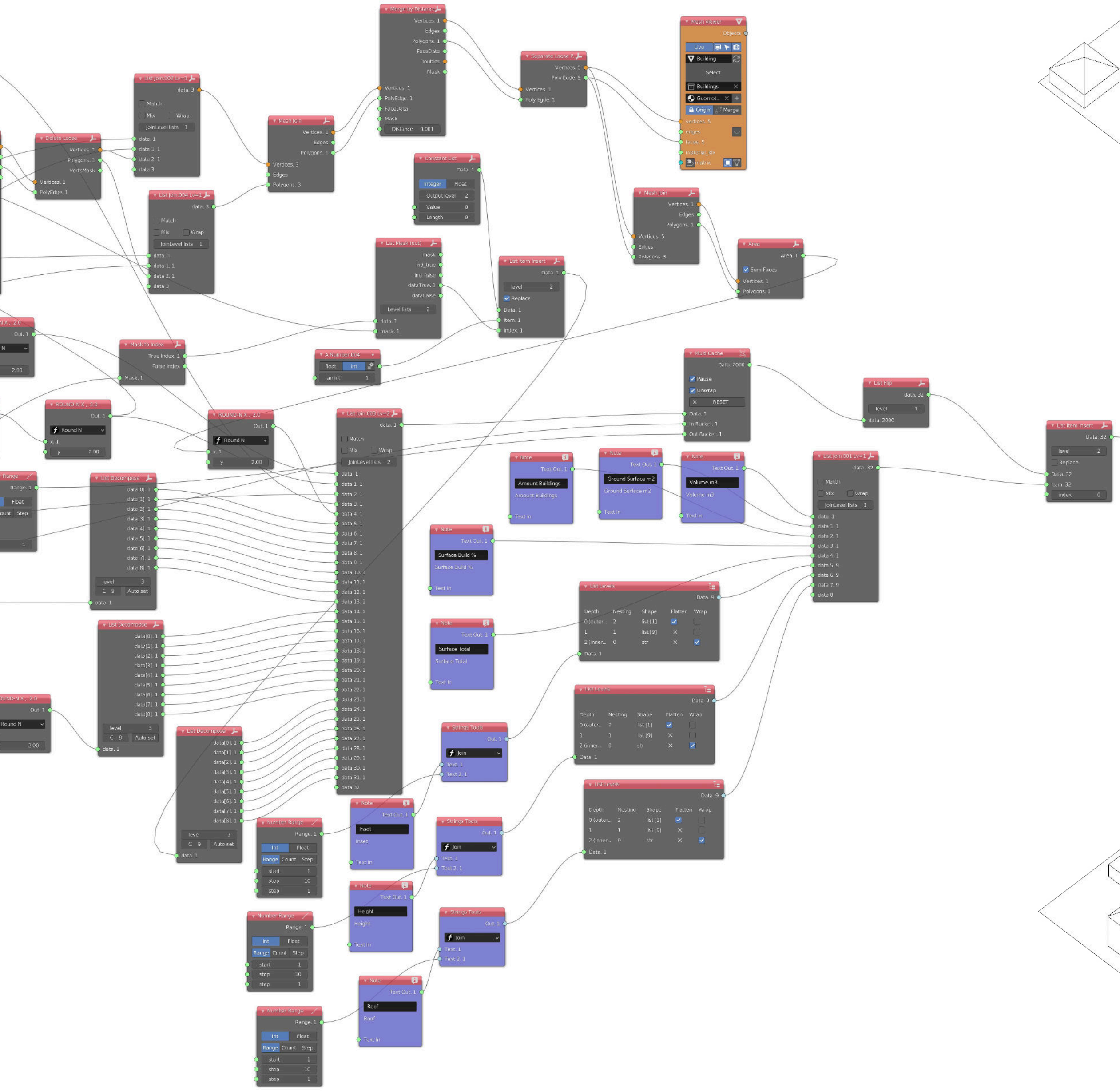
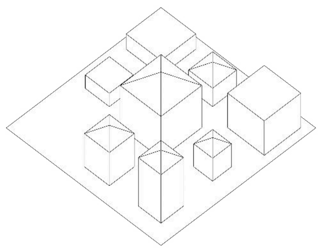
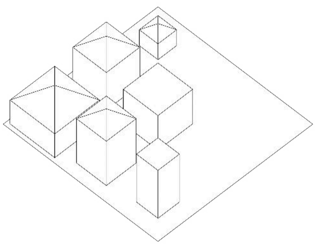
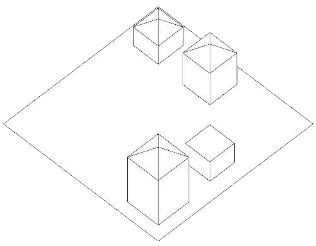
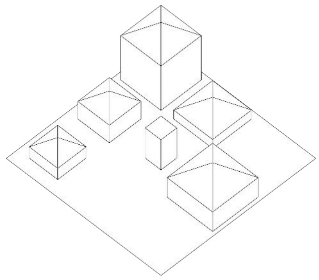
STEP 1 : PARAMETRIC GENERATION

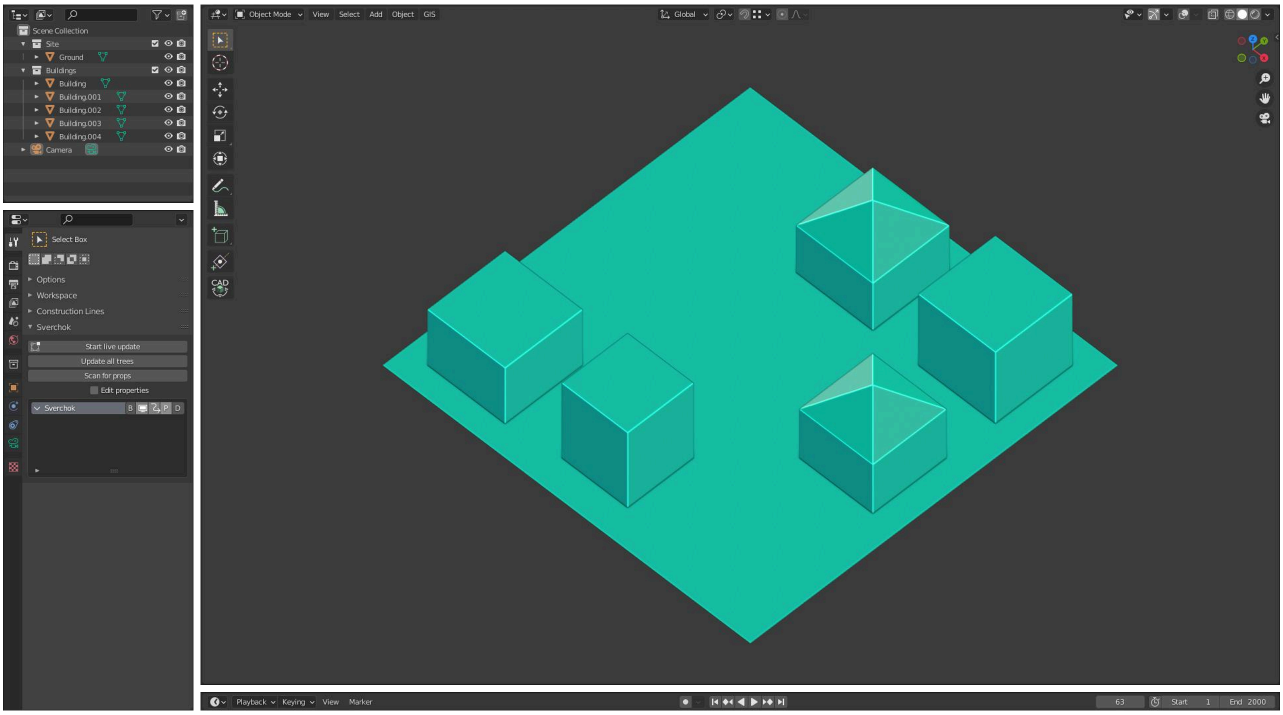
Notre climat sur terre n'est pas stable. Il y a une variation naturelle entre le jour et la nuit, d'un mois à l'autre, l'été et l'hiver, et à plus grande échelle, de la période glaciaire aux périodes interglaciaires. Il s'agit de fluctuations dues à des processus biologiques et physiques équilibrés.

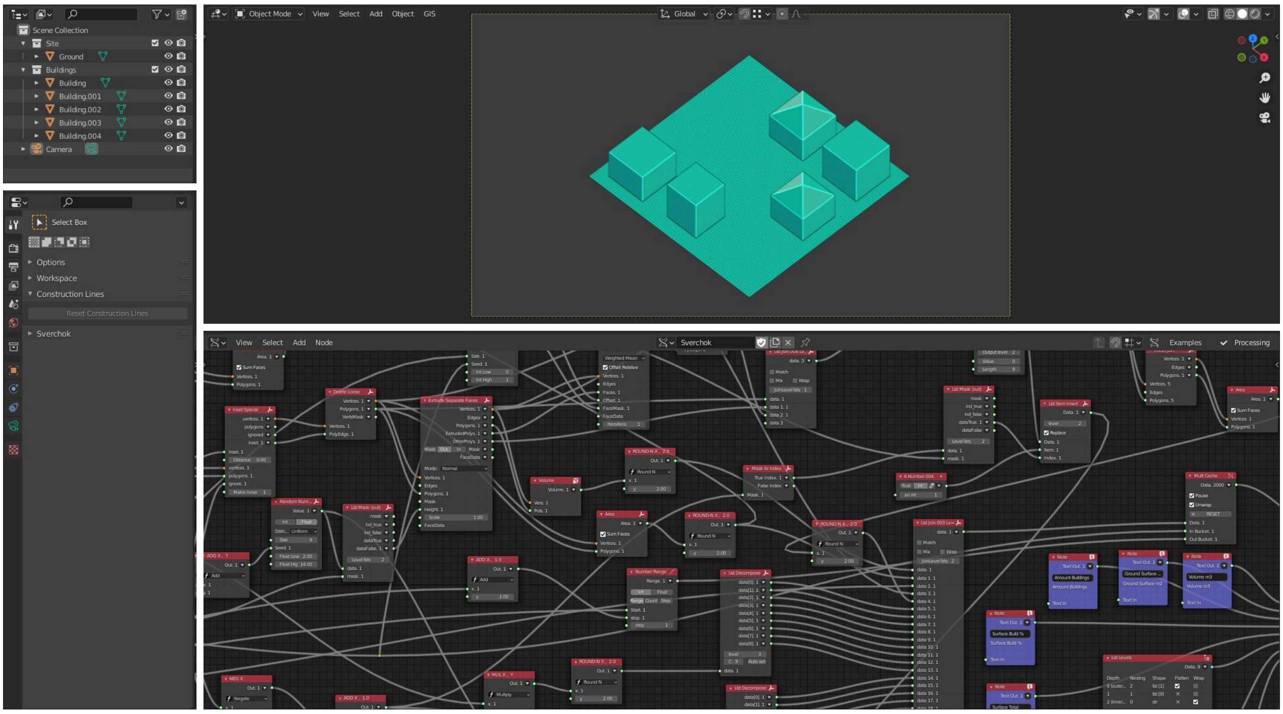
Depuis la révolution industrielle du 18e siècle, l'augmentation de la production et du nombre de la population mondiale a montré des effets déséquilibrants qui se rajoutent aux variations naturelles. Le climat mondial tend à s'extremiser, favorisé par des réactions chimiques en chaîne. Aujourd'hui, la gravité de la situation qui menace l'existence de l'homme sur terre ne fait aucun doute. Le monde scientifique décrit trois principes généraux pour contrer cette tendance : l'adaptation, la mitigation ou la transformation.

Depuis le début du millénaire, nous savons avec certitude que le changement climatique est une réalité incontestable, avec des conséquences pour la planète et les hommes qui nécessitent d'urgence un changement de notre comportement. Il est clair que l'homme doit s'adapter aux changements environnementaux. Mais est-ce suffisant ?
L'espèce humaine est dotée d'un pouvoir d'adaptation extraordinaire et unique, mais pour pouvoir contrer ces tendances qui menacent la vie sur terre, il est nécessaire de poursuivre des stratégies de mitigation voire de transformation pour inverser les dégâts causés au fil des dernières décennies et revenir à un état d'équilibre. Ces objectifs sont certainement exigeants et nécessitent des approches radicalement différentes de la politique environnementale actuelle. D'une manière générale, ils nous présentent trois stratégies de transformation : la sofficience, l'efficience et la consistance.
Le secteur de la construction est l'un des principaux émetteurs de carbone dans l'atmosphère, bien que des concepts tels que la construction passive montrent que cela n'est pas nécessaire et qu'il est possible de réduire la consommation d'énergie primaire non-renouvlablede manière drastique en utilisant des techniques *low-tech*. Ces approches alternatives en architecture sont basées sur l'optimisation de l'utilisation des sources d'énergie renouvelables gratuites. C'est-à-dire principalement les gains solaires. En réduisant la consommation d'énergie fossile ou fissile, et en compensenant le reste par l'utilisation et la transformation de l'énergie solaire en énergie utile pour le chauffage ou l'électricité, les bâtiments passifs ont un bilan énergétique très réduit, voire positif. Les stratégies de la passivité sont très diverses et une explication approfondie risque de dépasser le cadre de ce travail. C'est pourquoi je me concentre sur l'utilisation des apports gratuits par le rayonnement solaire. En effet, l'utilisation optimale de l'énergie solaire est celle qui ne passe pas par des équipements intermédiaires de transformation d'énergie comme les panneaux photovoltaïques, car ils sont toujours liés à des pertes d'un certain pourcentage causées par l'énergie grise ou l'inefficacité du mécanisme. Dans les années à venir, le nombre de jours d'ensoleillement va augmenter, ce qui signifie que nos maisons seront exposées au rayonnement solaire pendant de plus longues périodes. Cela est certainement avantageux en hiver pour réduire les besoins de chauffage mais devient un problème en mi-saison et en été, amplifié par l'augmentation de l'épaisseur d'isolation de nos maisons ces dernières années. Conformément aux principes d'efficacité, il s'agit donc d'augmenter les apports solaires en hiver et inversement en été de réduire le risque de surchauffe afin de ne pas perdre les économies d'énergie par une climatisation intensive. En outre, les rayons du soleil ne fournissent pas seulement de la chaleur par le biais du rayonnement infrarouge, mais aussi de la lumière visible.





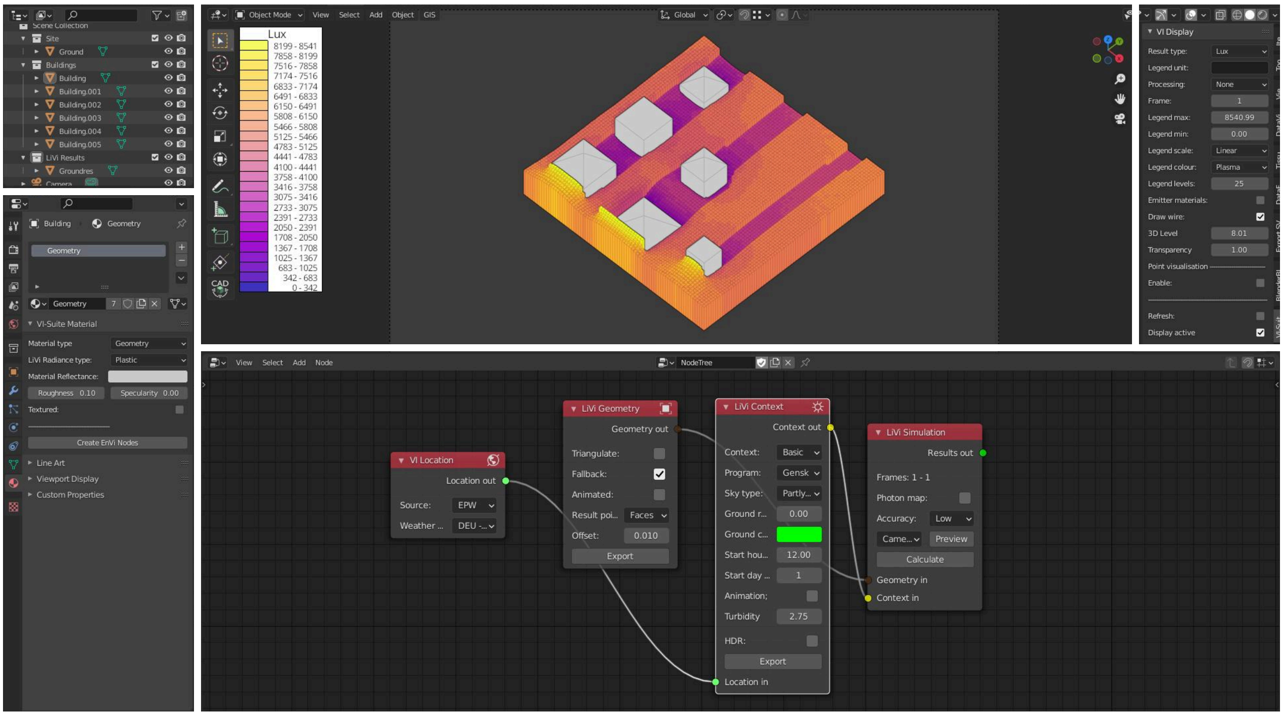
Volume m3	Ground Surface m2	Amount Buildings	Surface Build %	Surface Total	Inset 1	Inset 2	Inset 3	Inset 4	Inset 5	Inset 6	Inset 7	Inset 8	Inset 9	Height 1	Height 2	Height 3	Height 4	Height 5	Height 6	Height 7	Height 8	Height 9	Roof 1	Roof 2	Roof 3	Roof 4	Roof 5	Roof 6	Roof 7	Roof 8	Roof 9
1367.59	263.25	6	16.5	1441.95	0.55	0.73	0.3	0.48	0.39	0	0.41	0	0	7.23	2.31	8.6	7.22	7.04	0	4.46	0	0	1	1	0	0	1	0	1	0	0
3341.44	451.49	9	28.2	2833.24	0.56	0.32	0.63	0.56	0.55	0.5	0.42	0.67	0.48	8.61	10.5	5.49	8.13	12.72	12.76	3.51	4.49	2.62	0	1	1	0	0	1	0	1	0
5369.87	594.99	9	37.2	3780.04	0.63	0.72	0.47	0.61	0.84	0.84	0.38	0.42	0.33	13.6	8.57	13.67	10.58	10.37	4.59	13.72	2.07	5.04	0	0	1	1	0	0	0	1	1
4733.07	618.5	8	38.7	3449.81	0.88	0.63	0.88	0.73	0.72	0.43	0	0.3	0.45	4.66	12.45	4.48	13.02	7.86	9.34	0	8.22	5.56	0	0	1	1	1	0	0	1	0
2252.42	310.57	4	19.4	1669.99	0	0	0	0	0.59	0.67	0.76	0.61	0	0	0	0	0	3.29	9.14	8.36	7.03	0	0	0	0	0	1	0	1	1	0
1563.91	290.37	4	18.1	1386.62	0.84	0.5	0	0	0	0	0.62	0.55	0	2.92	11.36	0	0	0	0	8.01	2.86	0	0	1	0	0	0	0	1	0	0
3390.77	363.82	5	22.7	2314.56	0	0.77	0.56	0.73	0	0.62	0	0	0.46	0	13.62	12.43	8.37	0	2.14	0	0	8.27	0	1	0	1	0	0	0	0	1
1729.54	261.86	4	16.4	1362.83	0	0	0.82	0	0.44	0	0.56	0.54	0	0	0	7.95	0	3.71	0	7.02	4.98	0	0	0	1	0	0	0	1	1	0
1638.7	228.41	6	14.3	1530.38	0.31	0.6	0.6	0	0.39	0	0	0.45	0.35	11.26	2.25	9.6	0	7.98	0	0	11.13	4.03	0	0	0	0	1	0	0	0	0
2562.93	384.81	5	24.1	1985.58	0.76	0	0.68	0	0.6	0.43	0	0.76	0	4.16	0	7.56	0	7.04	7.83	0	7.85	0	1	0	1	0	0	1	0	0	0
149.81	62.39	1	3.9	214.67	0	0	0	0	0	0	0	0.59	0	0	0	0	0	0	0	0	2.4	0	0	0	0	0	0	0	0	1	0
4636.51	490.08	7	30.6	3235.47	0.39	0	0.46	0	0.31	0.85	0.84	0.32	0.87	11.33	0	11.89	0	13.67	7.44	9.31	11.31	9.7	1	0	1	0	0	1	1	0	1
1679.15	299.81	3	18.7	1347.1	0	0	0	0.88	0	0.57	0	0.77	0	0	0	0	2.1	0	13.49	0	5.82	0	0	0	0	0	0	0	0	0	0
3860.57	661.85	9	41.4	3132.56	0.61	0.76	0.82	0.3	0.49	0.87	0.61	0.49	0.62	12.19	4.15	2.65	6.34	5.3	8.36	5.67	5.65	3.34	1	0	0	0	1	1	0	0	0
2299.46	426.42	9	26.7	2231.88	0.81	0.41	0.33	0.52	0.47	0.62	0.48	0.48	0.37	4.68	8.28	8.61	2.55	6.33	4.68	10.26	3.96	2.84	0	1	0	1	1	0	0	1	1
2206.91	291	6	18.2	1820.7	0.43	0	0.63	0	0.52	0	0.71	0.4	0.34	5.54	0	4.3	0	11.44	0	9.65	8.91	2.47	1	0	0	0	1	0	1	1	1
922.11	101.41	2	6.3	677.07	0	0	0	0	0	0	0.68	0	0.32	0	0	0	0	0	0	9.99	0	5.08	0	0	0	0	0	0	1	0	1
2404.58	298.51	4	18.7	1695.51	0	0.6	0	0.41	0.81	0	0.7	0	0	0	11.13	0	3.66	5.98	0	10.06	0	0	0	0	0	1	0	0	1	0	0
2867.63	314.27	6	19.6	2155.94	0.36	0.76	0.45	0	0.5	0.35	0.7	0	0	9.06	12.77	12.7	0	2.43	10.3	6.54	0	0	1	0	1	0	0	0	1	0	0
1449.44	376.15	4	23.5	1409.44	0	0.84	0	0.79	0	0.72	0.53	0	0	0	5.47	0	2.26	0	2.61	5.63	0	0	0	1	0	0	0	1	1	0	0
2833.13	392.94	9	24.6	2411.59	0.33	0.47	0.73	0.31	0.42	0.33	0.48	0.7	0.48	4.5	7.78	7.05	12.31	4.05	6.07	5.25	10.29	4.64	1	1	0	0	0	0	0	1	0
2898.69	312.69	6	19.5	2198.33	0.43	0.59	0.55	0	0.4	0.5	0	0.71	0	8.21	13.36	11.19	0	4.65	10.23	0	6.71	0	1	0	0	0	0	0	0	0	0
4702.59	392.77	4	24.5	2721.87	0.61	0.87	0.76	0	0	0.71	0	0	0	13.52	10.39	14	0	0	10.88	0	0	0	1	0	0	0	0	1	0	0	0
1329.98	268.29	3	16.8	1127.21	0	0	0.9	0	0	0.74	0	0	0.38	0	0	5.35	0	0	3.41	0	0	8.67	0	0	0	0	0	1	0	0	0
2858.43	335.44	5	21	2127.29	0.82	0.65	0.47	0.41	0	0	0	0	0.63	5.7	8.23	11.22	11.47	0	0	0	0	10.87	0	0	0	1	0	0	0	0	1
4164.62	453.13	6	28.3	2864.7	0	0	0.76	0	0.82	0.41	0.46	0.6	0.74	0	0	10.82	0	6.6	13.75	12.72	4.52	10.9	0	0	1	0	0	0	0	1	0
3447.8	426.75	4	26.7	2237.8	0	0.79	0.74	0.82	0	0	0	0	0.75	0	8.73	3.5	6.77	0	0	0	0	13.46	0	1	0	0	0	0	0	0	1
1976.3	201.74	2	12.6	1239.88	0.74	0	0	0	0.77	0	0	0	0	12.37	0	0	0	7.43	0	0	0	0	1	0	0	0	1	0	0	0	0
3966.77	491.14	6	30.7	2765.77	0.82	0.47	0	0.76	0.57	0.63	0	0	0.76	9.73	6.57	0	3.96	13.55	6.16	0	0	9.03	1	1	0	0	1	0	0	0	0
4413.3	525.57	6	32.8	2937.05	0	0.53	0.7	0	0.88	0	0.9	0.44	0.65	0	13.5	11.24	0	4.5	0	12.9	2.82	2.9	0	1	1	0	1	0	1	0	0
2585.41	289.13	3	18.1	1631.53	0	0	0	0.89	0	0	0.85	0.34	0	0	0	0	13.47	0	0	3.21	13.14	0	0	0	0	0	0	0	1	0	0
5937.42	743.74	8	46.5	3923.47	0.82	0.52	0	0.87	0.74	0.79	0.36	0.86	0.67	4.98	7.4	0	5.12	12.44	4.22	2.24	13.44	10.17	1	1	0	1	0	0	0	0	1
1602.4	278.37	5	17.4	1508.28	0	0.57	0.55	0	0.82	0.41	0.31	0	0	0	11.36	3.11	0	2.17	13.23	7.12	0	0	0	0	1	0	0	0	0	0	0
748.08	112.61	2	7	661.77	0	0	0	0.68	0	0	0	0	0.41	0	0	0	5.33	0	0	0	0	10.19	0	0	0	1	0	0	0	0	0
3835.54	458.98	8	28.7	2877.73	0	0.49	0.44	0.47	0.79	0.37	0.68	0.47	0.71	0	9.22	13.43	3.72	11.69	6.36	4.24	5.11	9.56	0	1	1	0	1	1	0	1	0
2978.17	460.35	6	28.8	2257.9	0	0.66	0.87	0.39	0.78	0	0.41	0	0.68	0	7.57	4.31	8.98	9.44	0	3.24	0	5.38	0	1	0	0	0	0	1	0	0
3681.13	402.26	6	25.1	2641.79	0.87	0.58	0.42	0.65	0.67	0	0.36	0	0	6.62	12.32	13.33	10.43	9.6	0	4.4	0	0	1	1	0	1	0	0	1	0	0
623.29	81.77	2	5.1	568.88	0	0	0	0	0	0	0.42	0.53	0	0	0	0	0	0	0	7.57	7.66	0	0	0	0	0	0	0	1	1	0
494.55	59.46	1	3.7	388.82	0	0	0	0	0	0	0.58	0	0	0	0	0	0	0	0	8.32	0	0	0	0	0	0	0	0	1	0	0
3426.28	421.89	7	26.4	2425.53	0.54	0.33	0.77	0.47	0.57	0.48	0	0	0.77	5.01	2.55	10.12	2.52	3.4	9.25	0	0	13.01	0	1	1	1	0	0	0	0	0
301.95	77.99	1	4.9	292.74	0	0	0	0	0	0.66	0	0	0	0	0	0	0	0	3.87	0	0	0	0	0	0	0	0	0	0	0	0
2848.03	433.05	7	27.1	2304.34	0.52	0.87	0.74	0.66	0.39	0.39	0.33	0	0	3.38	9.31	3.6	4.89	5.93	12.31	9.99	0	0	0	1	0	0	0	1	0	0	0
1892.78	285.01	5	17.8	1665.88	0.37	0.67	0.38	0	0	0	0.7	0.62	0	12.02	3.26	10.94	0	0	0	6.73	6.91	0	0	0	1	0	0	0	1	1	0
2482.62	301.1	5	18.8	1788.08	0.8	0.36	0	0	0.52	0	0.54	0	0.61	13.87	8.59	0	0	7.33	0	2.58	0	3.39	0	1	0	0	1	0	1	0	1
2738.53	261.89	4	16.4	1786.7	0.89	0.63	0	0.35	0	0	0	0.4	0	11.41	9.62	0	11.1	0	0	0	7.29	0	1	0	0	1	0	0	0	0	0
4485.1	520.37	6	32.5	2997.68	0.77	0.68	0	0.75	0.49	0.86	0	0.56	0	3.36	13.69	0	6.22	10.49	11.6	0	6.98	0	1	1	0	0	0	0	0	1	0
6493.2	714.21	8	44.6	4164.53	0	0.88	0.74	0.51	0.72	0.78	0.69	0.55	0.72	0	12.7	5.42	5.59	11.5	5.89	12.38	7.37	8.58	0	1	0	1	0	0	1	0	1
790.22	70.32	1	4.4	517.57	0	0	0	0	0	0	0	0	0.63	0	0	0	0	0	0	0	0	11.24	0	0	0	0	0	0	0	0	0
5589.56	699.02	9	43.7	3968.78	0.48	0.45	0.86	0.83	0.71	0.64	0.63	0.43	0.76	7.94	4.74	5.07	6.76	6.53	13.96	6.9	11.26	11.13	0	1	0	1	0	0	0	0	1
223.53	36.53	1	2.3	221	0	0	0.45	0	0	0	0	0	0	0	0	6.12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3480.31	398.1	6	24.9	2377.8	0	0.33	0.51	0.69	0.47	0.87	0	0	0.65	0	2.31	4.53	9.42	3.18	9.44	0	0	13.77	0	0	1	1	0	1	0	0	1
3720.82	392.87	6	24.6	2553.57	0.79	0.32	0	0	0.36	0.67	0.33	0	0.89	12.16	8.73	0	0	9.03	8.43	11.93	0	7.74	1	0	0	0	1	1	1	0	0
3681.53	413.92	6	25.9	2654.13	0	0	0.57	0	0.65	0.62	0.8	0.46	0.59	0	0	4.22	0	2.1	13.63	11.62	11.09	10.06	0	0	1	0	1	1	1	1	1
2973.83	484.77	6	30.3	2332.6	0.55	0	0	0.61	0.31	0.88	0.78	0.75	0	3.12	0	0	4.91	8.37	5.43	12.35	2.49	0	1	0	0	1	0	1	1	1	0
3685.84	491.87	8	30.7	3002.7	0.36	0.88	0.59	0	0.62	0.47	0.82	0.32	0.37	13.81	6	10.08	0	6.25	11.76	4.97	7.5	12.53	1	0	1	0	0	1	1	1	1
3816.29	543.47	6	34	2804.8	0.89	0	0.7	0	0.51	0.79	0.45	0	0.83	3.05	0	6.93	0	8.79	8.54	11.69	0	8.27	1	0	0	0	1	0	0	0	1
3304.02	490.03	7	30.6	2556.71	0.35	0.44	0.55	0	0.64	0.63	0.78	0.85	0	6.38	7.41	7.95	0	8.86	5.48	5.71	6.5	0	1	0	1	0	0	1	0	0	0
1644.36	194.25	4	12.1	1331.38	0	0	0.6	0	0	0.47	0.49	0.52	0	0	0	12.4	0	0	5.38	5.08	8.77	0	0	0	1	0	0	0	1	1	0
483.43	60.97	2	3.8	483.2	0	0	0	0.35	0	0.47	0	0	0	0	0	0	9.99	0	6.78	0	0	0	0	0	0	1	0	0	0	0	0
806.98	66.18	2	4.1	671.51	0	0	0	0	0	0	0.53	0.31	0	0	0	0	0	0	0	13.08	9.55	0	0	0	0	0	0	0	1	1	0
3436.58	364.1	4	22.8	2087.63	0	0	0.83	0.6	0.88	0	0	0	0.48	0	0	12.15	6.94	9.58	0	0	0	4.88	0	0	1	0	0	0	0	0	0
1115.04	162.8	3	10.2	955.31	0	0	0	0.55	0	0.47	0.63	0	0	0	0	0	8.31	0	8	5.07	0	0	0	0	0	0	0	0	1	0	0
2259.85	326.46	5	20.4	1805.65	0.63	0.54	0	0	0	0.6	0	0.63	0.63	6.55	8.81	0	0	0	5.74	0	5.54	8.37	0	0	0	0	0	1	0	1	0
1754.14	302.2	5	18.9	1528.78	0	0.64	0.66	0	0	0.49	0.49	0	0.62	0	2.38	5.45	0	0	3.96	10.39	0	8.17	0	0	0	0	0	1	0	0	0
2189.99	284.16	6	17.8	1787.06	0	0.32	0.47	0.4	0	0.4	0	0.76	0.61	0	3.6	6.35	10.15	0	5.01	0	8.69	8.18	0	0	1	1	0	0	0	1	1
1989.39	213.6	5	13.3	1570.05	0.39	0	0.52	0	0.42	0.45	0	0.63	0	8.55	0	10.23	0	2.72	6.64	0	13.19	0	0	0	0	0	1	1	0	0	0
1788.42	273.96	4	17.1	1520.85	0	0	0.71	0	0.34	0	0.43	0.86	0	0	0	8.98	0	9.33	0	11.69	3.14	0	0	0	1	0	1	0	0	1	0
1823.64	211.73	3	13.2	1288.06	0	0	0	0	0.67	0.48	0	0	0.72	0	0	0	0	8.74	5.04	0	0	10.08	0	0	0	0	1	0	0	0	1
4530	434.79	7	27.2	3096.09	0.48	0.79	0.51	0.77	0.64	0.45	0.36	0	0	13.13	12.47	9.02	12.86	5.8	3.78	12.31	0	0	0	1	1	1	0	1	0	0	0
4911.11	575.74	7	36	3442.6	0.86	0	0.65	0.84	0.49	0.39	0.82	0.56	0	4.23	0	11.98	4.38	13	11.46	10.67	13.95	0	0	0	0	0	0	0	1	0	0
4961.56	671.32	8	42	3506.99	0.41	0.53	0.8	0.42	0.85	0.77	0.73	0	0.81	3.28	10.21	8.42	6.43	6.95	9.05	10.59	0	2.8	1	1	1	1	0	0	0	0	1
3721.26	503.5	9	31.5	3082.83	0.36	0.71	0.62	0.52	0.55	0.65	0.73	0.4	0.34	9.71	8.46	8.14	9.32	7.92	4.57	4.93	8.87	11.27	0	1	0	0	1	0	1	1	0
4226.12	448.31	7	28	3015.53	0	0.62	0.61	0.67	0.6	0.43	0.45	0	0.76	0	11.41	12.33	5.93	4.99	11.72	10.56	0	10.53	0	0	0	0	0	0	0	0	1
3327.12	323.48	5	20.2	2283.85	0.42	0	0.82	0.5	0.45	0	0	0	0.73	8.83	0	12.66	13.16	10.16	0	0	0	6.49	0	0	1	1	1	0	0	0	0
4708.85	665.83	9	41.6	3635.06	0.64	0.34	0.83	0.86	0.71	0.63	0.41	0.67	0.52	5.73	11.83	6.02	4.32	3.32	13.55	12.38	12.23	2.95	0	0	1	1	0	0	0	0	0
1486.69	161.65	2	10.1	1022.2	0	0	0.5	0	0	0	0	0	0.81	0	0	11.04	0	0	0	0	8.49	0	0	0	1	0	0	0	0	1	0
4383.17	589.34	8	36.8	3322.35	0.85	0.69	0.75	0	0.35	0.77	0.5	0.62	0.44	2.58	10.17	11.58	0	13.58	7.33	11.97	4.96	2.6	1	1	0	0	0	1	1	0	1
3247.55	430.5	6	26.9	2259.04	0.33	0.71	0	0.78	0.88	0.57	0	0	0.33	8.01	7.62	0	3.79	11.84	5.86	0	0	2.18	1	0	0	1	0	1	0	0	0
4511.77	538.14	8	33.6	3255.01	0.6	0.58	0	0.39	0.79	0.49	0.77	0.79	0.31	8.26	10.39	0	10.09	12.9	10.22	4.54	5.81	5.21	1	1	0	0	1	0	0	0	0
4102.17	509.97	7	31.9	2837.16	0	0.72	0	0.7	0.84	0.71	0.43	0.49	0.46	0	6.09	0	12.97	12.05	2.71	4.98	8.11	3.07	0	1	0	0	0	1	0	0	1
2014.47	248.73	4	15.5	1481.08	0.51	0	0	0	0.8	0	0	0.61	0.35	5.3	0	0	0	7.74	0	0	11.99	4.45	1	0	0	0	1	0	0	0	0
4086.2	435.28	4	27.2	2525.32	0	0	0.67	0.77	0	0.88	0	0.8	0	0	0	9.96	11.33	0	6.28	0	10.92	0	0	0	1	0	0	1	0	1	0
1775.81	242.53	3	15.2	1275.97	0	0	0	0.77	0	0	0	0.75	0.47	0	0	0	9.54	0	0	0	5.1	7.02	0	0	0	0	0	0	0	1	0

STEP 2 : SIMULATION AND ANALYSIS

La base de ce travail est basée sur l'analyse des mécanismes existants. Il s'agit d'étudier les différentes manières dont ils fonctionnent au niveau technique. L'avantage de se concentrer sur les structures cinétiques existantes est qu'il n'est pas nécessaire de s'appuyer sur des hypothèses simulées mais de pouvoir évaluer les résultats réels. La sélection des systèmes étudiés sera constituée de créations architecturales contemporaines et ne sera pas limitée à un certain pays et donc pas non plus à certains climats. Pour cette raison, l'analyse doit tenir compte des différents contextes.

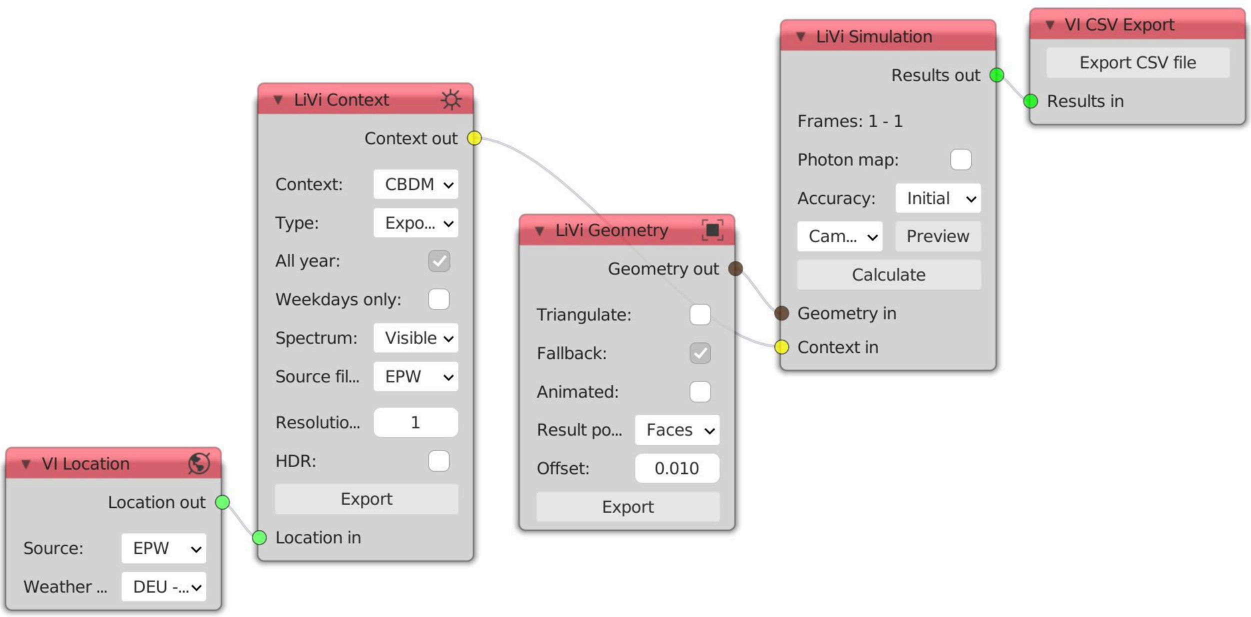
Afin d'évaluer l'utilité des exemples, différents critères doivent être pris en compte. Le premier est le niveau d'optimisation technique du mécanisme, ensuite il est important d'examiner l'utilité au niveau de l'utilisation architecturale. Ceci peut être accompagné d'une évaluation du caractère esthétique en fonction du contexte mais aussi de l'environnement vécu par l'habitant. Ce dernier point est bien sûr subjectif.

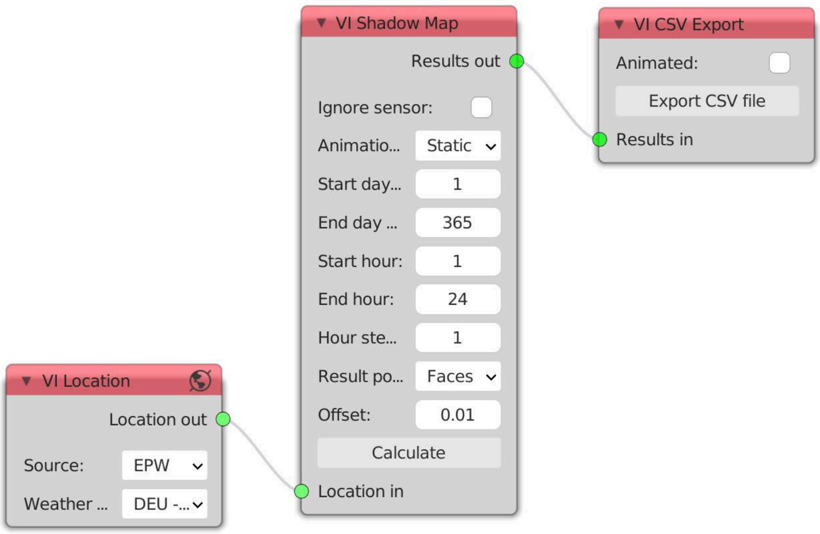
Ce travail d'analyse permettra de cristalliser les principales caractéristiques avantageuses pour les façades cinétiques et guidera donc la conception des mécanismes des étapes suivantes.

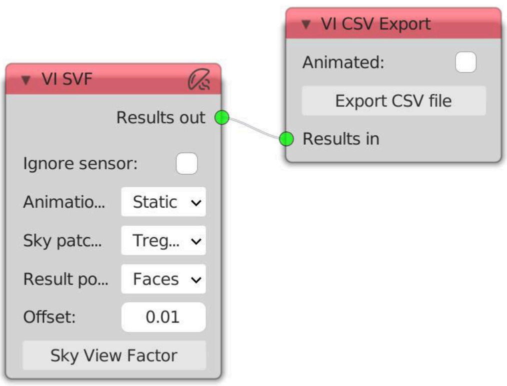


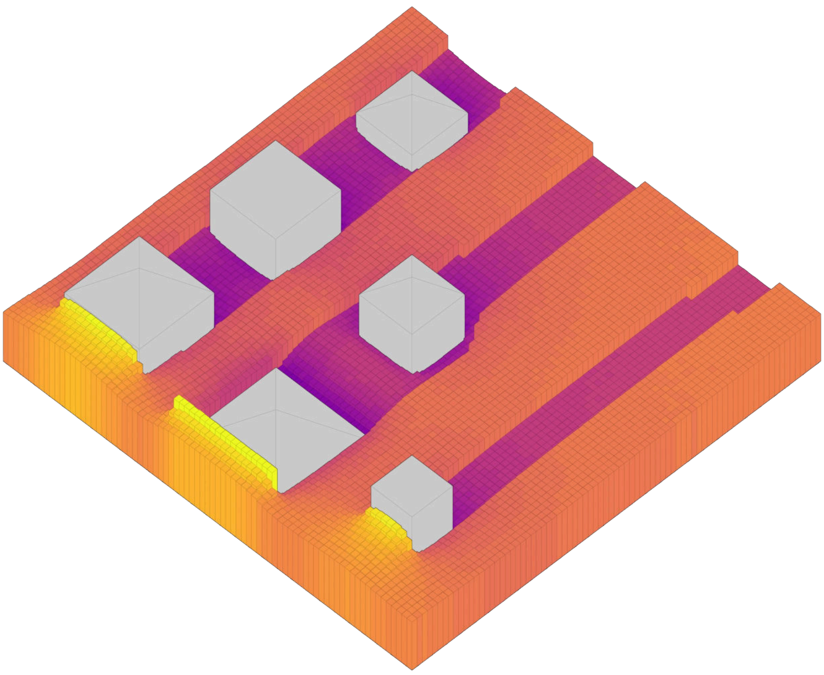
1 Ground X	1 Ground Y	1 Ground Z	1 Ground Sunlit %	2 Ground X	2 Ground Y	2 Ground Z	2 Ground Sunlit %	3 Ground X	3 Ground Y	3 Ground Z	3 Ground Sunlit %
-19.747	-19.747	0.01	100	-19.747	-19.747	0.01	100	-19.747	-19.747	0.01	100
-19.241	-19.747	0.01	100	-19.241	-19.747	0.01	100	-19.241	-19.747	0.01	100
-18.734	-19.747	0.01	100	-18.734	-19.747	0.01	100	-18.734	-19.747	0.01	100
-18.228	-19.747	0.01	100	-18.228	-19.747	0.01	100	-18.228	-19.747	0.01	100
-17.722	-19.747	0.01	100	-17.722	-19.747	0.01	100	-17.722	-19.747	0.01	100
-17.215	-19.747	0.01	100	-17.215	-19.747	0.01	100	-17.215	-19.747	0.01	100
-16.709	-19.747	0.01	100	-16.709	-19.747	0.01	100	-16.709	-19.747	0.01	100
-16.203	-19.747	0.01	100	-16.203	-19.747	0.01	100	-16.203	-19.747	0.01	100
-15.696	-19.747	0.01	100	-15.696	-19.747	0.01	100	-15.696	-19.747	0.01	100
-15.19	-19.747	0.01	100	-15.19	-19.747	0.01	100	-15.19	-19.747	0.01	100
-14.684	-19.747	0.01	100	-14.684	-19.747	0.01	100	-14.684	-19.747	0.01	99.312
-14.177	-19.747	0.01	100	-14.177	-19.747	0.01	100	-14.177	-19.747	0.01	98.875
-13.671	-19.747	0.01	100	-13.671	-19.747	0.01	99.438	-13.671	-19.747	0.01	98.875
-13.165	-19.747	0.01	98.875	-13.165	-19.747	0.01	98.875	-13.165	-19.747	0.01	98.438
-12.658	-19.747	0.01	98.875	-12.658	-19.747	0.01	98.875	-12.658	-19.747	0.01	97.125
-12.152	-19.747	0.01	98.875	-12.152	-19.747	0.01	98.875	-12.152	-19.747	0.01	96.5
-11.646	-19.747	0.01	98.875	-11.646	-19.747	0.01	97.875	-11.646	-19.747	0.01	96.062
-11.139	-19.747	0.01	97.375	-11.139	-19.747	0.01	97.062	-11.139	-19.747	0.01	95.938
-10.633	-19.747	0.01	96.75	-10.633	-19.747	0.01	96.5	-10.633	-19.747	0.01	95.938
-10.127	-19.747	0.01	96.312	-10.127	-19.747	0.01	96.125	-10.127	-19.747	0.01	95.938
-9.62	-19.747	0.01	96	-9.62	-19.747	0.01	95.938	-9.62	-19.747	0.01	95.438
-9.114	-19.747	0.01	95.938	-9.114	-19.747	0.01	95.938	-9.114	-19.747	0.01	94.875
-8.608	-19.747	0.01	95.938	-8.608	-19.747	0.01	95.938	-8.608	-19.747	0.01	94.5
-8.101	-19.747	0.01	95.938	-8.101	-19.747	0.01	95.938	-8.101	-19.747	0.01	94.25
-7.595	-19.747	0.01	95.938	-7.595	-19.747	0.01	95.562	-7.595	-19.747	0.01	94
-7.089	-19.747	0.01	95.312	-7.089	-19.747	0.01	95	-7.089	-19.747	0.01	93.875
-6.582	-19.747	0.01	94.875	-6.582	-19.747	0.01	94.625	-6.582	-19.747	0.01	93.688
-6.076	-19.747	0.01	94.562	-6.076	-19.747	0.01	94.375	-6.076	-19.747	0.01	93.562
-5.57	-19.747	0.01	94.375	-5.57	-19.747	0.01	94.188	-5.57	-19.747	0.01	93.438
-5.063	-19.747	0.01	94.125	-5.063	-19.747	0.01	94	-5.063	-19.747	0.01	93.312
-4.557	-19.747	0.01	94	-4.557	-19.747	0.01	93.875	-4.557	-19.747	0.01	93.25
-4.051	-19.747	0.01	93.875	-4.051	-19.747	0.01	93.75	-4.051	-19.747	0.01	93.125
-3.544	-19.747	0.01	93.75	-3.544	-19.747	0.01	93.625	-3.544	-19.747	0.01	93.062
-3.038	-19.747	0.01	93.625	-3.038	-19.747	0.01	93.5	-3.038	-19.747	0.01	93
-2.532	-19.747	0.01	93.5	-2.532	-19.747	0.01	93.438	-2.532	-19.747	0.01	92.938
-2.025	-19.747	0.01	93.438	-2.025	-19.747	0.01	93.312	-2.025	-19.747	0.01	92.875
-1.519	-19.747	0.01	93.312	-1.519	-19.747	0.01	93.25	-1.519	-19.747	0.01	92.812
-1.013	-19.747	0.01	93.25	-1.013	-19.747	0.01	93.188	-1.013	-19.747	0.01	92.75
-0.506	-19.747	0.01	93.188	-0.506	-19.747	0.01	93.125	-0.506	-19.747	0.01	92.688
0	-19.747	0.01	93.125	0	-19.747	0.01	93.062	0	-19.747	0.01	92.625
0.506	-19.747	0.01	93.062	0.506	-19.747	0.01	93	0.506	-19.747	0.01	92.625
1.013	-19.747	0.01	93	1.013	-19.747	0.01	92.938	1.013	-19.747	0.01	92.562
1.519	-19.747	0.01	92.938	1.519	-19.747	0.01	92.875	1.519	-19.747	0.01	92.5
2.025	-19.747	0.01	92.875	2.025	-19.747	0.01	92.812	2.025	-19.747	0.01	92.5

1 Ground X	1 Ground Y	1 Ground Z	1 Ground SVF	2 Ground X	2 Ground Y	2 Ground Z	2 Ground SVF	3 Ground X	3 Ground Y	3 Ground Z	3 Ground SVF	4 Ground X
-19.747	-19.747	0.01	86	-19.747	-19.747	0.01	84	-19.747	-19.747	0.01	80	-19.747
-19.241	-19.747	0.01	85	-19.241	-19.747	0.01	83	-19.241	-19.747	0.01	80	-19.241
-18.734	-19.747	0.01	84	-18.734	-19.747	0.01	82	-18.734	-19.747	0.01	80	-18.734
-18.228	-19.747	0.01	83	-18.228	-19.747	0.01	82	-18.228	-19.747	0.01	80	-18.228
-17.722	-19.747	0.01	84	-17.722	-19.747	0.01	82	-17.722	-19.747	0.01	80	-17.722
-17.215	-19.747	0.01	84	-17.215	-19.747	0.01	82	-17.215	-19.747	0.01	75	-17.215
-16.709	-19.747	0.01	80	-16.709	-19.747	0.01	77	-16.709	-19.747	0.01	72	-16.709
-16.203	-19.747	0.01	77	-16.203	-19.747	0.01	75	-16.203	-19.747	0.01	68	-16.203
-15.696	-19.747	0.01	75	-15.696	-19.747	0.01	73	-15.696	-19.747	0.01	67	-15.696
-15.19	-19.747	0.01	73	-15.19	-19.747	0.01	71	-15.19	-19.747	0.01	65	-15.19
-14.684	-19.747	0.01	71	-14.684	-19.747	0.01	68	-14.684	-19.747	0.01	64	-14.684
-14.177	-19.747	0.01	71	-14.177	-19.747	0.01	68	-14.177	-19.747	0.01	64	-14.177
-13.671	-19.747	0.01	68	-13.671	-19.747	0.01	68	-13.671	-19.747	0.01	59	-13.671
-13.165	-19.747	0.01	68	-13.165	-19.747	0.01	68	-13.165	-19.747	0.01	59	-13.165
-12.658	-19.747	0.01	69	-12.658	-19.747	0.01	68	-12.658	-19.747	0.01	60	-12.658
-12.152	-19.747	0.01	68	-12.152	-19.747	0.01	65	-12.152	-19.747	0.01	60	-12.152
-11.646	-19.747	0.01	68	-11.646	-19.747	0.01	66	-11.646	-19.747	0.01	60	-11.646
-11.139	-19.747	0.01	70	-11.139	-19.747	0.01	66	-11.139	-19.747	0.01	60	-11.139
-10.633	-19.747	0.01	71	-10.633	-19.747	0.01	68	-10.633	-19.747	0.01	60	-10.633
-10.127	-19.747	0.01	73	-10.127	-19.747	0.01	68	-10.127	-19.747	0.01	62	-10.127
-9.62	-19.747	0.01	79	-9.62	-19.747	0.01	68	-9.62	-19.747	0.01	63	-9.62
-9.114	-19.747	0.01	78	-9.114	-19.747	0.01	72	-9.114	-19.747	0.01	66	-9.114
-8.608	-19.747	0.01	78	-8.608	-19.747	0.01	70	-8.608	-19.747	0.01	65	-8.608
-8.101	-19.747	0.01	76	-8.101	-19.747	0.01	70	-8.101	-19.747	0.01	64	-8.101
-7.595	-19.747	0.01	77	-7.595	-19.747	0.01	71	-7.595	-19.747	0.01	64	-7.595
-7.089	-19.747	0.01	77	-7.089	-19.747	0.01	71	-7.089	-19.747	0.01	64	-7.089
-6.582	-19.747	0.01	75	-6.582	-19.747	0.01	71	-6.582	-19.747	0.01	62	-6.582
-6.076	-19.747	0.01	75	-6.076	-19.747	0.01	71	-6.076	-19.747	0.01	64	-6.076
-5.57	-19.747	0.01	75	-5.57	-19.747	0.01	71	-5.57	-19.747	0.01	62	-5.57
-5.063	-19.747	0.01	74	-5.063	-19.747	0.01	73	-5.063	-19.747	0.01	62	-5.063
-4.557	-19.747	0.01	71	-4.557	-19.747	0.01	73	-4.557	-19.747	0.01	60	-4.557
-4.051	-19.747	0.01	68	-4.051	-19.747	0.01	73	-4.051	-19.747	0.01	58	-4.051
-3.544	-19.747	0.01	68	-3.544	-19.747	0.01	73	-3.544	-19.747	0.01	58	-3.544
-3.038	-19.747	0.01	67	-3.038	-19.747	0.01	73	-3.038	-19.747	0.01	58	-3.038
-2.532	-19.747	0.01	68	-2.532	-19.747	0.01	74	-2.532	-19.747	0.01	58	-2.532
-2.025	-19.747	0.01	68	-2.025	-19.747	0.01	73	-2.025	-19.747	0.01	57	-2.025
-1.519	-19.747	0.01	68	-1.519	-19.747	0.01	73	-1.519	-19.747	0.01	57	-1.519
-1.013	-19.747	0.01	68	-1.013	-19.747	0.01	73	-1.013	-19.747	0.01	57	-1.013
-0.506	-19.747	0.01	68	-0.506	-19.747	0.01	71	-0.506	-19.747	0.01	57	-0.506
0	-19.747	0.01	68	0	-19.747	0.01	71	0	-19.747	0.01	57	0
0.506	-19.747	0.01	68	0.506	-19.747	0.01	70	0.506	-19.747	0.01	57	0.506
1.013	-19.747	0.01	68	1.013	-19.747	0.01	72	1.013	-19.747	0.01	57	1.013
1.519	-19.747	0.01	68	1.519	-19.747	0.01	73	1.519	-19.747	0.01	57	1.519
2.025	-19.747	0.01	69	2.025	-19.747	0.01	73	2.025	-19.747	0.01	57	2.025




VI Shadow Map
Results out
Ignore sensor:
Animatio...
Static
Start day...
1
End day ...
365
Start hour:
1
End hour:
24
Hour ste...
1
Result po...
Faces
Offset:
0.01
Calculate
Location in
VI CSV Export
Animated:
Export CSV file
Results in
VI Location
Location out
Source:
EPW
Weather ...
DEU -...


VI CSV Export
Animated:
Export CSV file
Results in
VI SVF
Results out
Ignore sensor:
Animatio...
Static
Sky patc...
Treg...
Result po...
Faces
Offset:
0.01
Sky View Factor



STEP 3 : DATA MANIPU-LATION

L'étape suivante consiste en l'élaboration du mécanisme au niveau de son fonctionnement cinétique. Pour ce faire, les résultats de la première étape sevent de base pour l'optimisation de la fonctionnalité. Il est d'importance d'examiner de multiples morphologies afin d'aboutir à une série de concepts variés.

L'expérimentation pour cette phase est avant tout une quête géométrique et mécanique. Ainsi, il est possible de faire varier la forme générale des modules autour de logiques géométriques simples, basées sur des formes triangulaires ou carrées. La réflexion sur le mouvement peut se développer en deux dimensions mais aussi en trois dimensions.

In [1]:

```python
import pandas as pd
import numpy as np
import pickle
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_validate
```

Data import and analysis

In [2]:

```python
base = pd.read_csv('./data/base.csv')
svf1 = pd.read_csv('./data/svf1.zip')
svf2 = pd.read_csv('./data/svf2.zip')
sl1 = pd.read_csv('./data/sl1.zip')
sl2 = pd.read_csv('./data/sl2.zip')
```

In [3]:

```python
display( base.head(), svf1.head(), svf2.head(), sl1.head(), sl2.head())
```

	Volume m3	Ground Surface m2	Amount Buildings	Surface Build %	Surface Total	Inset 1	Inset 2	Inset 3	Inset 4	Inset 5	...	Height 9	Roof 1	Roof 2	Roof 3	Roof 4	Roof 5	Roof 6	Roof 7	Roof 8	Roof 9
0	1367.59	263.25	6	16.5	1441.95	0.55	0.73	0.30	0.48	0.39	...	0.00	1	1	0	0	1	0	1	0	0
1	3341.44	451.49	9	28.2	2833.24	0.56	0.32	0.63	0.56	0.55	...	2.62	0	1	1	0	0	1	0	1	0
2	5369.87	594.99	9	37.2	3780.04	0.63	0.72	0.47	0.61	0.84	...	5.04	0	0	1	1	0	0	0	1	1
3	4733.07	618.50	8	38.7	3449.81	0.88	0.63	0.88	0.73	0.72	...	5.56	0	0	1	1	1	0	0	1	0
4	2252.42	310.57	4	19.4	1669.99	0.00	0.00	0.00	0.00	0.59	...	0.00	0	0	0	0	1	0	1	1	0

5 rows × 32 columns

	1 Ground X	1 Ground Y	1 Ground Z	1 Ground SVF	2 Ground X	2 Ground Y	2 Ground Z	2 Ground SVF	3 Ground X	3 Ground Y	...	998 Ground SVF	999 Ground X	999 Ground Y	999 Ground Z	999 Ground SVF	1000 Ground X	1000 Ground Y	1000 Ground Z	1000 Ground SVF	Unnamed: 4000
0	-19.747	-19.747	0.01	86.0	-19.747	-19.747	0.01	84.0	-19.747	-19.747	...	78.0	-19.747	-19.747	0.01	99.0	-19.747	-19.747	0.01	84.0	NaN
1	-19.241	-19.747	0.01	85.0	-19.241	-19.747	0.01	83.0	-19.241	-19.747	...	78.0	-19.241	-19.747	0.01	99.0	-19.241	-19.747	0.01	83.0	NaN
2	-18.734	-19.747	0.01	84.0	-18.734	-19.747	0.01	82.0	-18.734	-19.747	...	66.0	-18.734	-19.747	0.01	99.0	-18.734	-19.747	0.01	82.0	NaN
3	-18.228	-19.747	0.01	83.0	-18.228	-19.747	0.01	82.0	-18.228	-19.747	...	62.0	-18.228	-19.747	0.01	99.0	-18.228	-19.747	0.01	82.0	NaN
4	-17.722	-19.747	0.01	84.0	-17.722	-19.747	0.01	82.0	-17.722	-19.747	...	57.0	-17.722	-19.747	0.01	99.0	-17.722	-19.747	0.01	78.0	NaN

5 rows × 4001 columns

	1 Ground X	1 Ground Y	1 Ground Z	1 Ground SVF	2 Ground X	2 Ground Y	2 Ground Z	2 Ground SVF	3 Ground X	3 Ground Y	...	998 Ground SVF	999 Ground X	999 Ground Y	999 Ground Z	999 Ground SVF	1000 Ground X	1000 Ground Y	1000 Ground Z	1000 Ground SVF	Unnamed: 4000
0	-19.747	-19.747	0.01	95.0	-19.747	-19.747	0.01	87.0	-19.747	-19.747	...	80.0	-19.747	-19.747	0.01	96.0	-19.747	-19.747	0.01	86.0	NaN
1	-19.241	-19.747	0.01	95.0	-19.241	-19.747	0.01	87.0	-19.241	-19.747	...	80.0	-19.241	-19.747	0.01	96.0	-19.241	-19.747	0.01	84.0	NaN
2	-18.734	-19.747	0.01	95.0	-18.734	-19.747	0.01	87.0	-18.734	-19.747	...	80.0	-18.734	-19.747	0.01	95.0	-18.734	-19.747	0.01	84.0	NaN
3	-18.228	-19.747	0.01	95.0	-18.228	-19.747	0.01	86.0	-18.228	-19.747	...	78.0	-18.228	-19.747	0.01	95.0	-18.228	-19.747	0.01	83.0	NaN
4	-17.722	-19.747	0.01	95.0	-17.722	-19.747	0.01	84.0	-17.722	-19.747	...	78.0	-17.722	-19.747	0.01	95.0	-17.722	-19.747	0.01	83.0	NaN

5 rows × 4001 columns

	1 Ground X	1 Ground Y	1 Ground Z	1 Ground Sunlit %	2 Ground X	2 Ground Y	2 Ground Z	2 Ground Sunlit %	3 Ground X	3 Ground Y	...	998 Ground Sunlit %	999 Ground X	999 Ground Y	999 Ground Z	999 Ground Sunlit %	1000 Ground X	1000 Ground Y	1000 Ground Z	1000 Ground Sunlit %	Unnamed: 4000
0	-19.747	-19.747	0.01	100.0	-19.747	-19.747	0.01	100.0	-19.747	-19.747	...	100.000	-19.747	-19.747	0.01	100.0	-19.747	-19.747	0.01	100.0	NaN
1	-19.241	-19.747	0.01	100.0	-19.241	-19.747	0.01	100.0	-19.241	-19.747	...	100.000	-19.241	-19.747	0.01	100.0	-19.241	-19.747	0.01	100.0	NaN
2	-18.734	-19.747	0.01	100.0	-18.734	-19.747	0.01	100.0	-18.734	-19.747	...	100.000	-18.734	-19.747	0.01	100.0	-18.734	-19.747	0.01	100.0	NaN
3	-18.228	-19.747	0.01	100.0	-18.228	-19.747	0.01	100.0	-18.228	-19.747	...	98.875	-18.228	-19.747	0.01	100.0	-18.228	-19.747	0.01	100.0	NaN
4	-17.722	-19.747	0.01	100.0	-17.722	-19.747	0.01	100.0	-17.722	-19.747	...	96.438	-17.722	-19.747	0.01	100.0	-17.722	-19.747	0.01	100.0	NaN

5 rows × 4001 columns

	1 Ground X	1 Ground Y	1 Ground Z	1 Ground Sunlit %	2 Ground X	2 Ground Y	2 Ground Z	2 Ground Sunlit %	3 Ground X	3 Ground Y	...	998 Ground Sunlit %	999 Ground X	999 Ground Y	999 Ground Z	999 Ground Sunlit %	1000 Ground X	1000 Ground Y	1000 Ground Z	1000 Ground Sunlit %	Unnamed: 4000
0	-19.747	-19.747	0.01	100.0	-19.747	-19.747	0.01	100.0	-19.747	-19.747	...	100.0	-19.747	-19.747	0.01	100.0	-19.747	-19.747	0.01	100.0	NaN
1	-19.241	-19.747	0.01	100.0	-19.241	-19.747	0.01	100.0	-19.241	-19.747	...	100.0	-19.241	-19.747	0.01	100.0	-19.241	-19.747	0.01	100.0	NaN
2	-18.734	-19.747	0.01	100.0	-18.734	-19.747	0.01	100.0	-18.734	-19.747	...	100.0	-18.734	-19.747	0.01	100.0	-18.734	-19.747	0.01	100.0	NaN
3	-18.228	-19.747	0.01	100.0	-18.228	-19.747	0.01	100.0	-18.228	-19.747	...	100.0	-18.228	-19.747	0.01	100.0	-18.228	-19.747	0.01	100.0	NaN
4	-17.722	-19.747	0.01	100.0	-17.722	-19.747	0.01	100.0	-17.722	-19.747	...	100.0	-17.722	-19.747	0.01	100.0	-17.722	-19.747	0.01	100.0	NaN

5 rows × 4001 columns

In [4]:

```python
display(base.shape, svf1.shape, svf2.shape, sl1.shape, sl2.shape)
```

```
(2000, 32)
(6241, 4001)
(6241, 4001)
(6241, 4001)
(6241, 4001)
```

Data manipulation

Extraction and averaging of Shadowmap

In [5]:

```python
sl1_out = sl1.iloc[:, 3::4]
x = sl1_out[sl1_out > 1]
x.mean()
mean_sl1 = pd.DataFrame([x.mean()]).round(2)

sl2_out = sl2.iloc[:, 3::4]
x = sl2_out[sl2_out > 1]
x.mean()
mean_sl2 = pd.DataFrame([x.mean()]).round(2)

mean_sl_total = pd.concat([mean_sl1, mean_sl2], axis = 1)

mean_sl_total
```

Out[5]:

	1 Ground Sunlit %	2 Ground Sunlit %	3 Ground Sunlit %	4 Ground Sunlit %	5 Ground Sunlit %	6 Ground Sunlit %	7 Ground Sunlit %	8 Ground Sunlit %	9 Ground Sunlit %	10 Ground Sunlit %	...	991 Ground Sunlit %	992 Ground Sunlit %	993 Ground Sunlit %	994 Ground Sunlit %	995 Ground Sunlit %	996 Ground Sunlit %	997 Ground Sunlit %	998 Ground Sunlit %	999 Ground Sunlit %	1000 Ground Sunlit %
0	67.11	48.15	43.14	41.96	78.06	74.36	58.95	75.45	68.85	61.91	...	95.96	47.16	90.4	58.92	69.3	92.59	86.05	39.35	74.61	57.79

1 rows × 2000 columns

Extraction and averaging of SVF

In [6]:

```
svf1_out = svf1.iloc[:, 3::4]
x = svf1_out[svf1_out > 1]
x.mean()
mean_svf1 = pd.DataFrame([x.mean()]).round(2)

svf2_out = svf2.iloc[:, 3::4]
x = svf2_out[svf2_out > 1]
x.mean()
mean_svf2 = pd.DataFrame([x.mean()]).round(2)

mean_svf_total = pd.concat([mean_svf1, mean_svf2], axis = 1)

mean_svf_total
```

Out[6]:

	1 Ground SVF	2 Ground SVF	3 Ground SVF	4 Ground SVF	5 Ground SVF	6 Ground SVF	7 Ground SVF	8 Ground SVF	9 Ground SVF	10 Ground SVF	...	991 Ground SVF	992 Ground SVF	993 Ground SVF	994 Ground SVF	995 Ground SVF	996 Ground SVF	997 Ground SVF	998 Ground SVF	999 Ground SVF	1000 Ground SVF
0	74.36	53.28	50.73	53.24	74.81	78.38	66.91	77.81	72.78	66.77	...	95.16	48.86	88.71	52.07	71.2	85.21	88.42	45.87	73.52	63.62

1 rows × 2000 columns

Rotation and joining with Base table

In [7]:

```
mean_slx= mean_sl_total.T
mean_sl3 = mean_slx.rename(columns={0: "SL %"})

mean_svfx = mean_svf_total.T
mean_svf3 = mean_svfx.rename(columns={0: "SVF %"})

mean_svf4 = mean_svf3.reset_index(drop=True)
mean_sl4 = mean_sl3.reset_index(drop=True)

result = pd.concat([base, mean_sl4, mean_svf4],axis=1)
#result = result.drop_duplicates()
#result.to_csv('./results/result.csv')

result
```

Out[7]:

	Volume m3	Ground Surface m2	Amount Buildings	Surface Build %	Surface Total	Inset 1	Inset 2	Inset 3	Inset 4	Inset 5	...	Roof 2	Roof 3	Roof 4	Roof 5	Roof 6	Roof 7	Roof 8	Roof 9	SL %	SVF %
0	1367.59	263.25	6	16.5	1441.95	0.55	0.73	0.30	0.48	0.39	...	1	0	0	1	0	1	0	0	67.11	74.36
1	3341.44	451.49	9	28.2	2833.24	0.56	0.32	0.63	0.56	0.55	...	1	1	0	0	1	0	1	0	48.15	53.28
2	5369.87	594.99	9	37.2	3780.04	0.63	0.72	0.47	0.61	0.84	...	0	1	1	0	0	0	1	1	43.14	50.73
3	4733.07	618.50	8	38.7	3449.81	0.88	0.63	0.88	0.73	0.72	...	0	1	1	1	0	0	1	0	41.96	53.24
4	2252.42	310.57	4	19.4	1669.99	0.00	0.00	0.00	0.00	0.59	...	0	0	0	1	0	1	1	0	78.06	74.81
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1995	1984.40	211.23	2	13.2	1218.36	0.00	0.00	0.00	0.00	0.00	...	0	0	0	0	0	0	0	1	92.59	85.21
1996	1568.07	139.26	1	8.7	841.33	0.00	0.00	0.00	0.89	0.00	...	0	0	1	0	0	0	0	0	86.05	88.42
1997	5601.88	611.12	9	38.2	3986.92	0.64	0.76	0.45	0.76	0.51	...	1	1	0	1	0	0	0	1	39.35	45.87
1998	2643.47	392.71	4	24.5	1866.57	0.00	0.00	0.84	0.00	0.00	...	0	1	0	0	0	0	0	0	74.61	73.52
1999	2738.63	381.17	7	23.8	2281.72	0.64	0.64	0.59	0.00	0.53	...	0	0	0	1	0	1	0	0	57.79	63.62

2000 rows × 34 columns

Renaming

In [8]:

```
result.columns = range(result.shape[1])
clean_df = result.rename(columns={32: "sl", 33: "svf"})

clean_df
```

Out[8]:

	0	1	2	3	4	5	6	7	8	9	...	24	25	26	27	28	29	30	31	sl	svf
0	1367.59	263.25	6	16.5	1441.95	0.55	0.73	0.30	0.48	0.39	...	1	0	0	1	0	1	0	0	67.11	74.36
1	3341.44	451.49	9	28.2	2833.24	0.56	0.32	0.63	0.56	0.55	...	1	1	0	0	1	0	1	0	48.15	53.28
2	5369.87	594.99	9	37.2	3780.04	0.63	0.72	0.47	0.61	0.84	...	0	1	1	0	0	0	1	1	43.14	50.73
3	4733.07	618.50	8	38.7	3449.81	0.88	0.63	0.88	0.73	0.72	...	0	1	1	1	0	0	1	0	41.96	53.24
4	2252.42	310.57	4	19.4	1669.99	0.00	0.00	0.00	0.00	0.59	...	0	0	0	1	0	1	1	0	78.06	74.81
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1995	1984.40	211.23	2	13.2	1218.36	0.00	0.00	0.00	0.00	0.00	...	0	0	0	0	0	0	0	1	92.59	85.21
1996	1568.07	139.26	1	8.7	841.33	0.00	0.00	0.00	0.89	0.00	...	0	0	1	0	0	0	0	0	86.05	88.42
1997	5601.88	611.12	9	38.2	3986.92	0.64	0.76	0.45	0.76	0.51	...	1	1	0	1	0	0	0	1	39.35	45.87
1998	2643.47	392.71	4	24.5	1866.57	0.00	0.00	0.84	0.00	0.00	...	0	1	0	0	0	0	0	0	74.61	73.52
1999	2738.63	381.17	7	23.8	2281.72	0.64	0.64	0.59	0.00	0.53	...	0	0	0	1	0	1	0	0	57.79	63.62

2000 rows × 34 columns

Randomization

In [9]:

```
result1 = clean_df.dropna()
clean_df1 =result1.sample(frac=1)
clean_df2 = clean_df1.reset_index(drop=True)

clean_df2
```

Out[9]:

	0	1	2	3	4	5	6	7	8	9	...	24	25	26	27	28	29	30	31	sl	svf
0	2214.52	372.75	4	23.3	1702.37	0.00	0.66	0.86	0.35	0.00	...	0	1	1	0	0	1	0	0	70.92	76.31
1	5093.40	631.63	7	39.5	3454.74	0.00	0.59	0.80	0.72	0.78	...	0	1	0	0	0	0	1	1	52.40	52.49
2	2065.64	240.26	3	15.0	1377.61	0.00	0.30	0.76	0.82	0.00	...	0	0	0	0	0	0	0	0	71.62	78.67
3	3451.54	565.31	6	35.3	2546.07	0.84	0.00	0.80	0.83	0.72	...	0	1	1	0	0	1	0	0	53.92	61.16

	0	1	2	3	4	5	6	7	8	9	...	24	25	26	27	28	29	30	31	sl	svf
4	1647.92	233.72	3	14.6	1235.46	0.69	0.79	0.00	0.00	0.00	...	0	0	0	0	1	0	0	0	74.56	80.93
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1995	1644.38	168.43	3	10.5	1156.27	0.00	0.00	0.00	0.56	0.42	...	0	0	0	1	1	0	0	0	79.16	80.43
1996	313.35	75.42	1	4.7	295.16	0.00	0.00	0.00	0.00	0.00	...	0	0	0	0	0	0	0	0	94.23	93.81
1997	4288.60	567.39	8	35.5	3160.64	0.56	0.40	0.40	0.80	0.65	...	1	1	1	1	0	0	1	0	54.70	56.23
1998	3717.11	448.42	4	28.0	2360.91	0.00	0.00	0.73	0.73	0.00	...	0	0	1	0	0	0	0	0	68.32	69.13
1999	3021.50	306.38	3	19.1	1801.84	0.00	0.80	0.00	0.00	0.60	...	0	0	0	0	0	0	0	0	68.72	73.45

2000 rows × 34 columns

STEP 4 : MACHINE LEARNING

L'étude formelle des modules doit être suivie de recherches au niveau du mécanisme de régulation. C'est le mécanisme fondamental pour permettre la fonctionnalité des façades réesponsives. Le mode d'actionnement conditionne de manière significative la performance de l'ensemble.

La question du système de régulation implique l'étude de plusieurs autres facteurs : les différents stimuli de déclenchement, la force vectorielle et le mode de fonctionnement du mécanisme lui-même.

Machine learning

Predict SL

Multioutput

In [10]:

```
train = clean_df2[0:1999]

X = clean_df2.iloc[:,[-2,-1]]
y = clean_df2.iloc[:,:-2]

model = RandomForestRegressor()
model.fit(X, y)

data_in = [[70.65, 74.43]]
pred = model.predict(data_in)
m_prediction = pd.concat([pd.DataFrame(pred)] ,axis=1)
m_prediction
```

Out[10]:

	0	1	2	3	4	5	6	7	8	9	...	22	23	24	25	26	27	28	29	30	31
0	1896.1063	316.002	4.77	19.741	1575.7727	0.5944	0.0522	0.0441	0.6345	0.5629	...	2.0204	0.07	0.02	0.03	0.0	0.67	0.03	0.14	0.03	0.08

1 rows × 32 columns

In [11]:

```
#filename = 'trained_model.sav'
#pickle.dump(model, open(filename, 'wb'))
```

Feature Ranking

In [12]:

```
train = clean_df2[0:1950]
test = clean_df2[-50:]

y_train = train['sl']
X_train = train.iloc[:,:32]
y_test = test['sl']
X_test = test.iloc[:,:32]

regr = RandomForestRegressor()
regr.fit(X_train,y_train)
importances = regr.feature_importances_
std = np.std([tree.feature_importances_ for tree in regr.estimators_], axis=0)
indices = np.argsort(importances)[::-1]

print("Feature ranking:")

for f in range(X_train.shape[1]):
    print("%d. feature %d (%f)" % (f + 1, indices[f], importances[indices[f]]))

prediction = regr.predict(X_test)

y_test = y_test.reset_index(drop=True)
print("Mean absolute error for SL prediction:",mean_absolute_error(y_test, prediction))

res1 = pd.concat([pd.DataFrame(prediction, columns=['Predicted sl']),y_test],axis=1)
```

```
Feature ranking:
1. feature 4 (0.929833)
2. feature 2 (0.014546)
3. feature 15 (0.008225)
4. feature 5 (0.007657)
5. feature 6 (0.004766)
6. feature 14 (0.003985)
7. feature 13 (0.003894)
8. feature 12 (0.002953)
9. feature 7 (0.002752)
10. feature 22 (0.002681)
11. feature 16 (0.001939)
12. feature 0 (0.001903)
13. feature 11 (0.001779)
14. feature 21 (0.001687)
15. feature 8 (0.001683)
16. feature 20 (0.001424)
17. feature 17 (0.001352)
18. feature 1 (0.001167)
19. feature 18 (0.001003)
20. feature 3 (0.000994)
21. feature 9 (0.000962)
22. feature 10 (0.000811)
23. feature 19 (0.000750)
24. feature 31 (0.000207)
25. feature 30 (0.000156)
26. feature 23 (0.000148)
27. feature 26 (0.000148)
28. feature 24 (0.000128)
29. feature 28 (0.000124)
30. feature 25 (0.000118)
31. feature 27 (0.000115)
32. feature 29 (0.000112)
Mean absolute error for SL prediction: 1.4926340000000025
```

Mean Error for i Features

In [13]:

```
train = clean_df2[0:1950]
test = clean_df2[-50:]

for i in range(1,len(train.columns)-2):
    chosen_columns = indices[0:i]
```

```
y_train = train['sl']
X_train = train.iloc[:,chosen_columns]
y_test = test['sl']
X_test = test.iloc[:,chosen_columns]

regr = RandomForestRegressor()
regr.fit(X_train,y_train)
prediction = regr.predict(X_test)

y_test = y_test.reset_index(drop=True)
print("Mean absolute error for SL prediction using TOP ",i," columns: ",mean_absolute_error(y_test, prediction))
```

```
Mean absolute error for SL prediction using TOP  1  columns:  3.29798133333332
Mean absolute error for SL prediction using TOP  2  columns:  2.893278000000009
Mean absolute error for SL prediction using TOP  3  columns:  2.8330699999999935
Mean absolute error for SL prediction using TOP  4  columns:  2.386773999999993
Mean absolute error for SL prediction using TOP  5  columns:  2.1040099999999944
Mean absolute error for SL prediction using TOP  6  columns:  2.1904879999999944
Mean absolute error for SL prediction using TOP  7  columns:  1.816751999999994
Mean absolute error for SL prediction using TOP  8  columns:  1.7960899999999915
Mean absolute error for SL prediction using TOP  9  columns:  1.5612639999999864
Mean absolute error for SL prediction using TOP  10  columns:  1.585743999999994
Mean absolute error for SL prediction using TOP  11  columns:  1.5068019999999942
Mean absolute error for SL prediction using TOP  12  columns:  1.6455759999999922
Mean absolute error for SL prediction using TOP  13  columns:  1.4339079999999933
Mean absolute error for SL prediction using TOP  14  columns:  1.4810379999999896
Mean absolute error for SL prediction using TOP  15  columns:  1.566459999999996
Mean absolute error for SL prediction using TOP  16  columns:  1.4828259999999964
Mean absolute error for SL prediction using TOP  17  columns:  1.431337999999993
Mean absolute error for SL prediction using TOP  18  columns:  1.4576340000000019
Mean absolute error for SL prediction using TOP  19  columns:  1.441774000000005
Mean absolute error for SL prediction using TOP  20  columns:  1.5403159999999998
Mean absolute error for SL prediction using TOP  21  columns:  1.4626139999999992
Mean absolute error for SL prediction using TOP  22  columns:  1.520925999999999
Mean absolute error for SL prediction using TOP  23  columns:  1.439647999999998
Mean absolute error for SL prediction using TOP  24  columns:  1.436219999999996
Mean absolute error for SL prediction using TOP  25  columns:  1.4806299999999928
Mean absolute error for SL prediction using TOP  26  columns:  1.4543060000000005
Mean absolute error for SL prediction using TOP  27  columns:  1.4413919999999996
Mean absolute error for SL prediction using TOP  28  columns:  1.4973679999999985
Mean absolute error for SL prediction using TOP  29  columns:  1.4068899999999962
Mean absolute error for SL prediction using TOP  30  columns:  1.5220819999999988
Mean absolute error for SL prediction using TOP  31  columns:  1.4333159999999996
```

Predict SVF

Feature Ranking

In [14]:
```
train = clean_df2[0:1950]
test = clean_df2[-50:]

y_train = train['svf']
X_train = train.iloc[:,:32]
y_test = test['svf']
X_test = test.iloc[:,:32]

regr = RandomForestRegressor()
regr.fit(X_train,y_train)
importances = regr.feature_importances_
std = np.std([tree.feature_importances_ for tree in regr.estimators_], axis=0)
indices = np.argsort(importances)[::-1]

print("Feature ranking:")

for f in range(X_train.shape[1]):
    print("%d. feature %d (%f)" % (f + 1, indices[f], importances[indices[f]]))

prediction = regr.predict(X_test)

y_test = y_test.reset_index(drop=True)
print("Mean absolute error for SVF prediction:",mean_absolute_error(y_test, prediction))
res2 = pd.concat([pd.DataFrame(prediction, columns=['Predicted svf']),y_test],axis=1)
```

```
Feature ranking:
1. feature 4 (0.969600)
2. feature 2 (0.019784)
3. feature 0 (0.001016)
4. feature 1 (0.000696)
5. feature 9 (0.000661)
6. feature 18 (0.000659)
7. feature 11 (0.000556)
8. feature 5 (0.000555)
9. feature 3 (0.000518)
10. feature 7 (0.000493)
11. feature 15 (0.000425)
12. feature 22 (0.000418)
13. feature 10 (0.000407)
14. feature 16 (0.000403)
15. feature 20 (0.000403)
16. feature 17 (0.000398)
17. feature 13 (0.000388)
18. feature 14 (0.000364)
19. feature 8 (0.000358)
20. feature 21 (0.000350)
21. feature 12 (0.000327)
22. feature 6 (0.000323)
23. feature 19 (0.000322)
24. feature 31 (0.000086)
25. feature 23 (0.000078)
26. feature 24 (0.000073)
27. feature 27 (0.000072)
28. feature 25 (0.000066)
29. feature 28 (0.000060)
30. feature 26 (0.000053)
31. feature 30 (0.000048)
32. feature 29 (0.000041)
Mean absolute error for SVF prediction: 1.1891019999999883
```

Mean Error for i Features

In [15]:
```
train = clean_df2[0:1950]
test = clean_df2[-50:]

for i in range(1,len(train.columns)-2):
    chosen_columns = indices[0:i]
```

```
y_train = train['svf']
X_train = train.iloc[:,chosen_columns]
y_test = test['svf']
X_test = test.iloc[:,chosen_columns]

regr = RandomForestRegressor()
regr.fit(X_train,y_train)
prediction = regr.predict(X_test)

y_test = y_test.reset_index(drop=True)
print("Mean absolute error for SVF prediction using TOP ",i," columns: ",mean_absolute_error(y_test, prediction))
```

```
Mean absolute error for SVF prediction using TOP  1  columns:  2.198972333333349
Mean absolute error for SVF prediction using TOP  2  columns:  1.3710119999999977
Mean absolute error for SVF prediction using TOP  3  columns:  1.2855479999999921
Mean absolute error for SVF prediction using TOP  4  columns:  1.275231999999991
Mean absolute error for SVF prediction using TOP  5  columns:  1.280377999999996
Mean absolute error for SVF prediction using TOP  6  columns:  1.2586679999999888
Mean absolute error for SVF prediction using TOP  7  columns:  1.2117879999999943
Mean absolute error for SVF prediction using TOP  8  columns:  1.1956779999999938
Mean absolute error for SVF prediction using TOP  9  columns:  1.2042119999999925
Mean absolute error for SVF prediction using TOP  10  columns:  1.2299279999999921
Mean absolute error for SVF prediction using TOP  11  columns:  1.2411619999999925
Mean absolute error for SVF prediction using TOP  12  columns:  1.2214219999999918
Mean absolute error for SVF prediction using TOP  13  columns:  1.1960639999999927
Mean absolute error for SVF prediction using TOP  14  columns:  1.1837239999999882
Mean absolute error for SVF prediction using TOP  15  columns:  1.1743079999999855
Mean absolute error for SVF prediction using TOP  16  columns:  1.1701919999999857
Mean absolute error for SVF prediction using TOP  17  columns:  1.170093999999992
Mean absolute error for SVF prediction using TOP  18  columns:  1.148175999999989
Mean absolute error for SVF prediction using TOP  19  columns:  1.142709999999987
Mean absolute error for SVF prediction using TOP  20  columns:  1.1415399999999898
Mean absolute error for SVF prediction using TOP  21  columns:  1.142739999999992
Mean absolute error for SVF prediction using TOP  22  columns:  1.1754219999999966
Mean absolute error for SVF prediction using TOP  23  columns:  1.1621579999999918
Mean absolute error for SVF prediction using TOP  24  columns:  1.1837739999999928
Mean absolute error for SVF prediction using TOP  25  columns:  1.1444039999999942
Mean absolute error for SVF prediction using TOP  26  columns:  1.1645279999999958
Mean absolute error for SVF prediction using TOP  27  columns:  1.1115119999999912
Mean absolute error for SVF prediction using TOP  28  columns:  1.1985799999999915
Mean absolute error for SVF prediction using TOP  29  columns:  1.162623999999994
Mean absolute error for SVF prediction using TOP  30  columns:  1.1597819999999976
Mean absolute error for SVF prediction using TOP  31  columns:  1.1886779999999897
```

Predictions SL SVF

In [16]:

```
prediction_table = pd.concat([res1, res2],axis=1)
print(prediction_table)
#prediction_table.to_csv('./results/predictions.csv')
```

	Predicted sl	sl	Predicted svf	svf
0	58.7837	59.18	64.7022	62.18
1	71.8761	73.69	75.7381	75.11
2	63.1055	64.94	64.2142	66.68
3	85.2611	86.60	84.0265	84.68
4	85.7951	85.61	85.9892	85.38
5	42.1787	42.14	45.9416	44.83
6	54.1991	55.05	59.7876	61.96
7	55.3532	56.65	62.7869	61.70
8	60.9393	57.90	66.8105	67.73
9	50.1140	49.39	55.0614	54.59
10	66.8116	63.49	71.0814	73.37
11	63.1807	59.39	67.9807	66.54
12	44.7185	45.02	48.3794	47.66
13	70.9782	71.16	77.7433	80.02
14	70.1116	70.09	74.6877	77.58
15	90.7376	91.45	91.6214	91.16
16	67.9070	65.64	66.4687	64.11
17	44.3331	39.61	51.4158	49.49
18	52.5456	49.13	55.3019	54.65
19	87.2450	87.44	90.1052	89.51
20	77.8817	78.53	77.8142	78.82
21	76.8034	75.81	81.2643	80.35
22	95.8815	96.33	93.7365	93.52
23	61.8811	62.45	64.0814	63.61
24	87.3445	87.62	89.8244	90.89
25	62.2802	59.92	65.5822	66.54
26	67.6007	67.67	74.3573	76.03
27	87.5500	89.55	87.7435	89.10
28	82.4820	84.01	84.4570	85.09
29	66.9675	66.54	69.2954	68.37
30	81.7892	82.96	85.0688	85.35
31	87.7493	87.47	89.5493	89.89
32	59.1931	64.47	62.5328	64.49
33	62.8286	61.84	65.1423	66.28
34	78.7683	77.20	76.9401	76.91
35	64.4328	63.81	69.3258	70.52
36	54.2822	50.99	57.0303	54.39
37	44.6765	45.35	47.8250	48.76
38	54.9299	51.78	58.7927	56.61
39	62.2771	63.49	63.6329	63.51
40	51.1927	49.38	58.8553	59.59
41	90.5980	88.98	91.4475	91.50
42	50.2257	48.60	53.5367	54.15
43	72.7624	70.30	76.7173	74.05
44	82.4048	80.48	80.4959	79.10
45	79.0168	79.16	80.8417	80.43
46	94.1863	94.23	93.9763	93.81
47	51.1636	54.70	53.5682	56.23
48	65.6803	68.32	68.4712	69.13
49	69.5416	68.72	75.2505	73.45

	Predicted sl	sl	Predicted svf	svf
0	79.5923	83.31	82.2418000000001	80.9
1	50.2079	48.61	51.5055	52.18
2	78.6392	78.83	82.2945	81.43
3	61.657	62.23	66.6192	69.44
4	64.2946	62.03	65.9299	65.24
5	49.429	49.39	55.0088	54.59
6	86.8902	86.01	88.1129000000001	88.68
7	64.0646	61.94	68.7594	68.6
8	85.1452	85.5	89.4201000000001	91.12
9	95.1836	94.77	95.9055000000001	95.67
10	86.3004	86.72	90.3246	91.49
11	75.5911	76.16	74.3451	74.43
12	59.893	60.72	63.3958	63.22
13	55.8868	51.51	58.8072	59.22
14	62.6631	62.45	64.7053	63.61
15	46.7265	44.05	50.8407	48.66
16	47.933	46.82	50.9167	50.28
17	46.2124	47.71	50.2118	52.35
18	50.2748	46.02	55.7168	56.2
19	75.9324999999999	76.72	76.1590000000001	75.57
20	62.1793	62.2	66.2909	66.44
21	43.6885	41.29	47.4322	47.11
22	89.4458	89.37	90.396	90.17
23	48.1475	45.56	46.0963	43.94
24	46.2781	47.66	49.4941	47.98
25	52.7982	55.58	58.1941	58.68
26	89.2827	88.98	89.2603	88.28
27	69.0284	74.45	71.0158999999999	70.53
28	55.7832	60.27	61.812	59.75
29	87.1349	86.65	89.432	90.55
30	91.6713	91.86	91.4221000000001	91.14
31	86.1624	85.21	85.5776999999999	85.13
32	85.8982	85.59	88.3469000000001	89.07
33	63.8572	64.94	64.1871	66.68
34	85.9750000000001	85.61	85.8584	85.38
35	51.1037	48.76	58.7357	58.7
36	62.4136	62.92	64.9444	65.54
37	39.5078	38.76	43.926	44.46
38	54.3057	54.2	57.7984	56.04
39	96.2773	96.79	94.5623000000001	94.92
40	83.7664	83.42	85.585	85.46
41	75.6302	81.28	77.3162	76.34
42	39.1848	40.87	44.7737	47.62
43	49.8086	50.74	54.0602	51.98
44	50.9818	51.2	51.5205	50.47
45	54.8997	60.88	57.1376	60.38
46	46.9706	47.34	54.9594	55.53
47	89.4243000000001	88.28	90.2071000000001	89.79
48	76.1796	74.99	81.1882	81.88
49	61.8397	56.6	61.7586	61.1

	Volume m3	Ground Surface m2	Amount Buildings	Surface Build %	Surface Total	Inset 1	Inset 2	Inset 3	Inset 4	Inset 5	Inset 6	Inset 7	Inset 8	Inset 9
0	1367.59	263.25	6	16.5	1441.95	0.55	0.73	0.3	0.48	0.39	0	0.41	0	0
1	3341.44	451.49	9	28.2	2833.24	0.56	0.32	0.63	0.56	0.55	0.5	0.42	0.67	0.48
2	5369.87	594.99	9	37.2	3780.04	0.63	0.72	0.47	0.61	0.84	0.84	0.38	0.42	0.33
3	4733.07	618.5	8	38.7	3449.81	0.88	0.63	0.88	0.73	0.72	0.43	0	0.3	0.45
4	2252.42	310.57	4	19.4	1669.99	0	0	0	0	0.59	0.67	0.76	0.61	0
5	1563.91	290.37	4	18.1	1386.62	0.84	0.5	0	0	0	0	0.62	0.55	0
6	3390.77	363.82	5	22.7	2314.56	0	0.77	0.56	0.73	0	0.62	0	0	0.46
7	1729.54	261.86	4	16.4	1362.83	0	0	0.82	0	0.44	0	0.56	0.54	0
8	1638.7	228.41	6	14.3	1530.38	0.31	0.6	0.6	0	0.39	0	0	0.45	0.35
9	2562.93	384.81	5	24.1	1985.58	0.76	0	0.68	0	0.6	0.43	0	0.76	0
10	149.81	62.39	1	3.9	214.67	0	0	0	0	0	0	0	0.59	0
11	4636.51	490.08	7	30.6	3235.47	0.39	0	0.46	0	0.31	0.85	0.84	0.32	0.87
12	1679.15	299.81	3	18.7	1347.1	0	0	0	0.88	0	0.57	0	0.77	0
13	3860.57	661.85	9	41.4	3132.56	0.61	0.76	0.82	0.3	0.49	0.87	0.61	0.49	0.62
14	2299.46	426.42	9	26.7	2231.88	0.81	0.41	0.33	0.52	0.47	0.62	0.48	0.48	0.37
15	2206.91	291	6	18.2	1820.7	0.43	0	0.63	0	0.52	0	0.71	0.4	0.34
16	922.11	101.41	2	6.3	677.07	0	0	0	0	0	0	0.68	0	0.32
17	2404.58	298.51	4	18.7	1695.51	0	0.6	0	0.41	0.81	0	0.7	0	0
18	2867.63	314.27	6	19.6	2155.94	0.36	0.76	0.45	0	0.5	0.35	0.7	0	0
19	1449.44	376.15	4	23.5	1409.44	0	0.84	0	0.79	0	0.72	0.53	0	0
20	2833.13	392.94	9	24.6	2411.59	0.33	0.47	0.73	0.31	0.42	0.33	0.48	0.7	0.48
21	2898.69	312.69	6	19.5	2198.33	0.43	0.59	0.55	0	0.4	0.5	0	0.71	0
22	4702.59	392.77	4	24.5	2721.87	0.61	0.87	0.76	0	0	0.71	0	0	0
23	1329.98	268.29	3	16.8	1127.21	0	0	0.9	0	0	0.74	0	0	0.38
24	2858.43	335.44	5	21	2127.29	0.82	0.65	0.47	0.41	0	0	0	0	0.63
25	4164.62	453.13	6	28.3	2864.7	0	0	0.76	0	0.82	0.41	0.46	0.6	0.74
26	3447.8	426.75	4	26.7	2237.8	0	0.79	0.74	0.82	0	0	0	0	0.75
27	1976.3	201.74	2	12.6	1239.88	0.74	0	0	0	0.77	0	0	0	0
28	3966.77	491.14	6	30.7	2765.77	0.82	0.47	0	0.76	0.57	0.63	0	0	0.76
29	4413.3	525.57	6	32.8	2937.05	0	0.53	0.7	0	0.88	0	0.9	0.44	0.65
30	2585.41	289.13	3	18.1	1631.53	0	0	0	0.89	0	0	0.85	0.34	0
31	5937.42	743.74	8	46.5	3923.47	0.82	0.52	0	0.87	0.74	0.79	0.36	0.86	0.67
32	1602.4	278.37	5	17.4	1508.28	0	0.57	0.55	0	0.82	0.41	0.31	0	0
33	748.08	112.61	2	7	661.77	0	0	0	0.68	0	0	0	0	0.41
34	3835.54	458.98	8	28.7	2877.73	0	0.49	0.44	0.47	0.79	0.37	0.68	0.47	0.71
35	2978.17	460.35	6	28.8	2257.9	0	0.66	0.87	0.39	0.78	0	0.41	0	0.68
36	3681.13	402.26	6	25.1	2641.79	0.87	0.58	0.42	0.65	0.67	0	0.36	0	0
37	623.29	81.77	2	5.1	568.88	0	0	0	0	0	0	0.42	0.53	0
38	494.55	59.46	1	3.7	388.82	0	0	0	0	0	0	0.58	0	0
39	3426.28	421.89	7	26.4	2425.53	0.54	0.33	0.77	0.47	0.57	0.48	0	0	0.77
40	301.95	77.99	1	4.9	292.74	0	0	0	0	0	0.66	0	0	0
41	2848.03	433.05	7	27.1	2304.34	0.52	0.87	0.74	0.66	0.39	0.39	0.33	0	0
42	1892.78	285.01	5	17.8	1665.88	0.37	0.67	0.38	0	0	0	0.7	0.62	0
43	2482.62	301.1	5	18.8	1788.08	0.8	0.36	0	0	0.52	0	0.54	0	0.61
44	2738.53	261.89	4	16.4	1786.7	0.89	0.63	0	0.35	0	0	0	0.4	0
45	4485.1	520.37	6	32.5	2997.68	0.77	0.68	0	0.75	0.49	0.86	0	0.56	0
46	6493.2	714.21	8	44.6	4164.53	0	0.88	0.74	0.51	0.72	0.78	0.69	0.55	0.72
47	790.22	70.32	1	4.4	517.57	0	0	0	0	0	0	0	0	0.63
48	5589.56	699.02	9	43.7	3968.78	0.48	0.45	0.86	0.83	0.71	0.64	0.63	0.43	0.76
49	223.53	36.53	1	2.3	221	0	0	0.45	0	0	0	0	0	0
50	3480.31	398.1	6	24.9	2377.8	0	0.33	0.51	0.69	0.47	0.87	0	0	0.65
51	3720.82	392.87	6	24.6	2553.57	0.79	0.32	0	0	0.36	0.67	0.33	0	0.89
52	3681.53	413.92	6	25.9	2654.13	0	0	0.57	0	0.65	0.62	0.8	0.46	0.59
53	2973.83	484.77	6	30.3	2332.6	0.55	0	0	0.61	0.31	0.88	0.78	0.75	0
54	3685.84	491.87	8	30.7	3002.7	0.36	0.88	0.59	0	0.62	0.47	0.82	0.32	0.37
55	3816.29	543.47	6	34	2804.8	0.89	0	0.7	0	0.51	0.79	0.45	0	0.83
56	3304.02	490.03	7	30.6	2556.71	0.35	0.44	0.55	0	0.64	0.63	0.78	0.85	0
57	1644.36	194.25	4	12.1	1331.38	0	0	0.6	0	0	0.47	0.49	0.52	0
58	483.43	60.97	2	3.8	483.2	0	0	0	0.35	0	0.47	0	0	0
59	806.98	66.18	2	4.1	671.51	0	0	0	0	0	0	0.53	0.31	0
60	3436.58	364.1	4	22.8	2087.63	0	0	0.83	0.6	0.88	0	0	0	0.48
61	1115.04	162.8	3	10.2	955.31	0	0	0	0.55	0	0.47	0.63	0	0
62	2259.85	326.46	5	20.4	1805.65	0.63	0.54	0	0	0	0.6	0	0.63	0.63
63	1754.14	302.2	5	18.9	1528.78	0	0.64	0.66	0	0	0.49	0.49	0	0.62
64	2189.99	284.16	6	17.8	1787.06	0	0.32	0.47	0.4	0	0.4	0	0.76	0.61
65	1989.39	213.6	5	13.3	1570.05	0.39	0	0.52	0	0.42	0.45	0	0.63	0
66	1788.42	273.96	4	17.1	1520.85	0	0	0.71	0	0.34	0	0.43	0.86	0
67	1823.64	211.73	3	13.2	1288.06	0	0	0	0	0.67	0.48	0	0	0.72
68	4530	434.79	7	27.2	3096.09	0.48	0.79	0.51	0.77	0.64	0.45	0.36	0	0
69	4911.11	575.74	7	36	3442.6	0.86	0	0.65	0.84	0.49	0.39	0.82	0.56	0
70	4961.56	671.32	8	42	3506.99	0.41	0.53	0.8	0.42	0.85	0.77	0.73	0	0.81
71	3721.26	503.5	9	31.5	3082.83	0.36	0.71	0.62	0.52	0.55	0.65	0.73	0.4	0.34
72	4226.12	448.31	7	28	3015.53	0	0.62	0.61	0.67	0.6	0.43	0.45	0	0.76
73	3327.12	323.48	5	20.2	2283.85	0.42	0	0.82	0.5	0.45	0	0	0	0.73
74	4708.85	665.83	9	41.6	3635.06	0.64	0.34	0.83	0.86	0.71	0.63	0.41	0.67	0.52
75	1486.69	161.65	2	10.1	1022.2	0	0	0.5	0	0	0	0	0.81	0
76	4383.17	589.34	8	36.8	3322.35	0.85	0.69	0.75	0	0.35	0.77	0.5	0.62	0.44
77	3247.55	430.5	6	26.9	2259.04	0.33	0.71	0	0.78	0.88	0.57	0	0	0.33
78	4511.77	538.14	8	33.6	3255.01	0.6	0.58	0	0.39	0.79	0.49	0.77	0.79	0.31
79	4102.17	509.97	7	31.9	2837.16	0	0.72	0	0.7	0.84	0.71	0.43	0.49	0.46
80	2014.47	248.73	4	15.5	1481.08	0.51	0	0	0	0.8	0	0	0.61	0.35

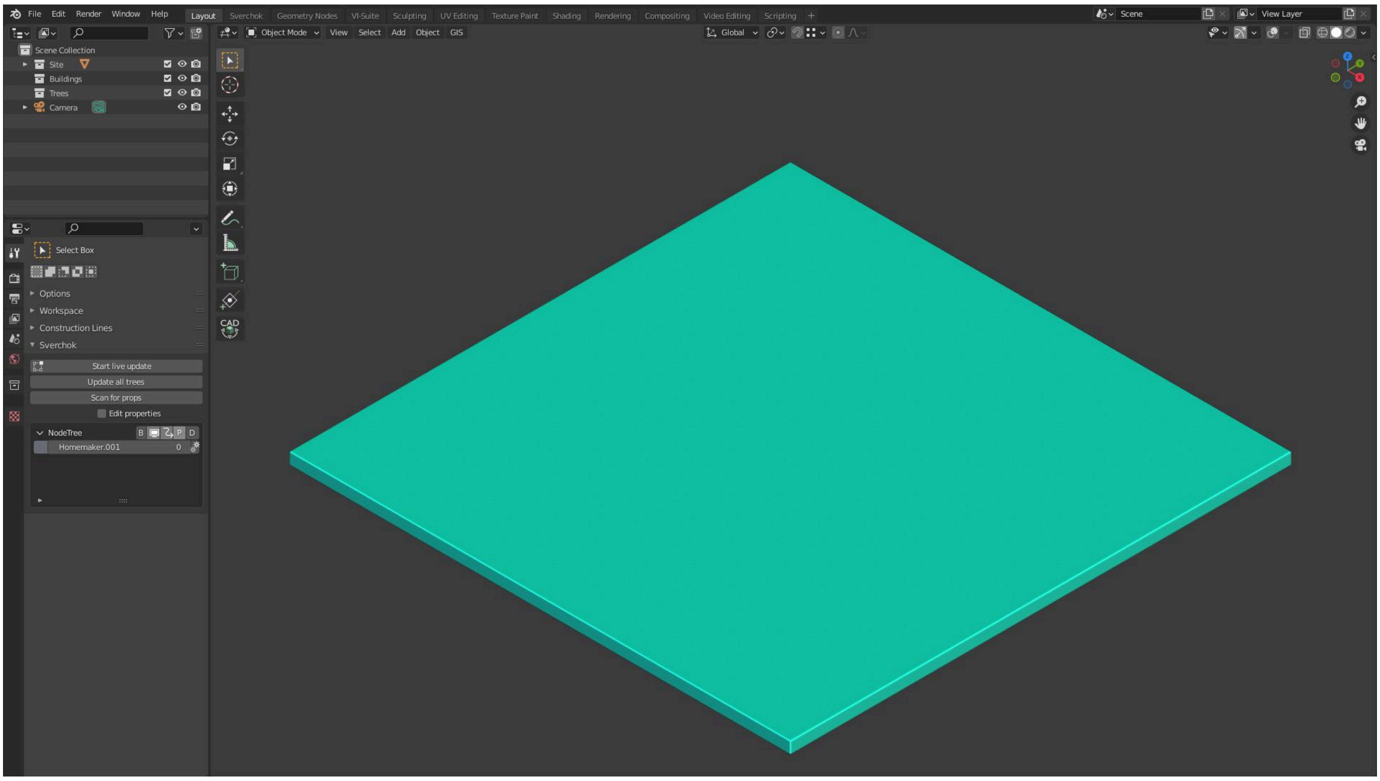
	Height 1	Height 2	Height 3	Height 4	Height 5	Height 6	Height 7	Height 8	Height 9	Roof 1	Roof 2	Roof 3	Roof 4	Roof 5	Roof 6	Roof 7	Roof 8	Roof 9	SL %	SVF %
0	7.23	2.31	8.6	7.22	7.04	0	4.46	0	0	1	1	0	0	1	0	1	0	0	67.11	74.36
1	8.61	10.5	5.49	8.13	12.72	12.76	3.51	4.49	2.62	0	1	1	0	0	1	0	1	0	48.15	53.28
2	13.6	8.57	13.67	10.58	10.37	4.59	13.72	2.07	5.04	0	0	1	1	0	0	0	1	1	43.14	50.73
3	4.66	12.45	4.48	13.02	7.86	9.34	0	8.22	5.56	0	0	1	1	1	0	0	1	0	41.96	53.24
4	0	0	0	0	3.29	9.14	8.36	7.03	0	0	0	0	0	1	0	1	1	0	78.06	74.81
5	2.92	11.36	0	0	0	0	8.01	2.86	0	0	1	0	0	0	0	1	0	0	74.36	78.38
6	0	13.62	12.43	8.37	0	2.14	0	0	8.27	0	1	0	1	0	0	0	0	1	58.95	66.91
7	0	0	7.95	0	3.71	0	7.02	4.98	0	0	0	1	0	0	0	1	1	0	75.45	77.81
8	11.26	2.25	9.6	0	7.98	0	0	11.13	4.03	0	0	0	0	1	0	0	0	0	68.85	72.78
9	4.16	0	7.56	0	7.04	7.83	0	7.85	0	1	0	1	0	0	1	0	0	0	61.91	66.77
10	0	0	0	0	0	0	0	2.4	0	0	0	0	0	0	0	0	1	0	96.24	95.32
11	11.33	0	11.89	0	13.67	7.44	9.31	11.31	9.7	1	0	1	0	0	1	1	0	1	61.79	57.81
12	0	0	0	2.1	0	13.49	0	5.82	0	0	0	0	0	0	0	0	0	0	79.61	78.47
13	12.19	4.15	2.65	6.34	5.3	8.36	5.67	5.65	3.34	1	0	0	0	1	1	0	0	0	43.03	50.01
14	4.68	8.28	8.61	2.55	6.33	4.68	10.26	3.96	2.84	0	1	0	1	1	0	0	1	1	53.88	59.61
15	5.54	0	4.3	0	11.44	0	9.65	8.91	2.47	1	0	0	0	1	0	1	1	1	67.58	68.59
16	0	0	0	0	0	0	9.99	0	5.08	0	0	0	0	0	0	1	0	1	91.74	89.39
17	0	11.13	0	3.66	5.98	0	10.06	0	0	0	0	0	1	0	0	1	0	0	68.99	73.62
18	9.06	12.77	12.7	0	2.43	10.3	6.54	0	0	1	0	1	0	0	0	1	0	0	59.83	67.94
19	0	5.47	0	2.26	0	2.61	5.63	0	0	0	1	0	0	0	1	1	0	0	70.32	76.32
20	4.5	7.78	7.05	12.31	4.05	6.07	5.25	10.29	4.64	1	1	0	0	0	0	0	1	0	55.16	58.1
21	8.21	13.36	11.19	0	4.65	10.23	0	6.71	0	1	0	0	0	0	0	0	0	0	58.85	65.41
22	13.52	10.39	14	0	0	10.88	0	0	0	1	0	0	0	0	1	0	0	0	58.38	71.23
23	0	0	5.35	0	0	3.41	0	0	8.67	0	0	0	0	0	1	0	0	0	83.48	85.13
24	5.7	8.23	11.22	11.47	0	0	0	0	10.87	0	0	0	1	0	0	0	0	1	61.18	69.1
25	0	0	10.82	0	6.6	13.75	12.72	4.52	10.9	0	0	1	0	0	0	0	1	0	63.8	61.91
26	0	8.73	3.5	6.77	0	0	0	0	13.46	0	1	0	0	0	0	0	0	1	63.12	67.5
27	12.37	0	0	0	7.43	0	0	0	0	1	0	0	0	1	0	0	0	0	76.06	81.9
28	9.73	6.57	0	3.96	13.55	6.16	0	0	9.03	1	1	0	0	1	0	0	0	0	55.66	61.16
29	0	13.5	11.24	0	4.5	0	12.9	2.82	2.9	0	1	1	0	1	0	1	0	0	58.24	60.31
30	0	0	0	13.47	0	0	3.21	13.14	0	0	0	0	0	0	0	1	0	0	81.08	80.65
31	4.98	7.4	0	5.12	12.44	4.22	2.24	13.44	10.17	1	1	0	1	0	0	0	0	1	48.18	49.81
32	0	11.36	3.11	0	2.17	13.23	7.12	0	0	0	0	1	0	0	0	0	0	0	71.1	75.64
33	0	0	0	5.33	0	0	0	0	10.19	0	0	0	1	0	0	0	0	0	87.92	87.92
34	0	9.22	13.43	3.72	11.69	6.36	4.24	5.11	9.56	0	1	1	0	1	1	0	1	0	55.09	56.03
35	0	7.57	4.31	8.98	9.44	0	3.24	0	5.38	0	1	0	0	0	0	1	0	0	58.15	63.48
36	6.62	12.32	13.33	10.43	9.6	0	4.4	0	0	1	1	0	1	0	0	1	0	0	54.47	65.59
37	0	0	0	0	0	0	7.57	7.66	0	0	0	0	0	0	0	1	1	0	91.96	89.52
38	0	0	0	0	0	0	8.32	0	0	0	0	0	0	0	0	1	0	0	94.58	93.71
39	5.01	2.55	10.12	2.52	3.4	9.25	0	0	13.01	0	1	1	1	0	0	0	0	0	59.92	64.45
40	0	0	0	0	0	3.87	0	0	0	0	0	0	0	0	0	0	0	0	94.44	93.96
41	3.38	9.31	3.6	4.89	5.93	12.31	9.99	0	0	0	1	0	0	0	1	0	0	0	54.37	64.84
42	12.02	3.26	10.94	0	0	0	6.73	6.91	0	0	0	1	0	0	0	1	1	0	70.39	72.99
43	13.87	8.59	0	0	7.33	0	2.58	0	3.39	0	1	0	0	1	0	1	0	1	65.67	72.16
44	11.41	9.62	0	11.1	0	0	0	7.29	0	1	0	0	1	0	0	0	0	0	67.56	76.65
45	3.36	13.69	0	6.22	10.49	11.6	0	6.98	0	1	1	0	0	0	0	0	1	0	49.21	57.23
46	0	12.7	5.42	5.59	11.5	5.89	12.38	7.37	8.58	0	1	0	1	0	0	1	0	1	43.58	46.15
47	0	0	0	0	0	0	0	0	11.24	0	0	0	0	0	0	0	0	0	95.89	92.56
48	7.94	4.74	5.07	6.76	6.53	13.96	6.9	11.26	11.13	0	1	0	1	0	0	0	0	1	39.48	43.34
49	0	0	6.12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	94.47	95.47
50	0	2.31	4.53	9.42	3.18	9.44	0	0	13.77	0	0	1	1	0	1	0	0	1	64.94	66.68
51	12.16	8.73	0	0	9.03	8.43	11.93	0	7.74	1	0	0	0	1	1	1	0	0	60.74	64.04
52	0	0	4.22	0	2.1	13.63	11.62	11.09	10.06	0	0	1	0	1	1	1	1	1	69.85	65.14
53	3.12	0	0	4.91	8.37	5.43	12.35	2.49	0	1	0	0	1	0	1	1	1	0	65.4	65.62
54	13.81	6	10.08	0	6.25	11.76	4.97	7.5	12.53	1	0	1	0	0	1	1	1	1	49.54	55.59
55	3.05	0	6.93	0	8.79	8.54	11.69	0	8.27	1	0	0	0	1	0	0	0	1	56.46	59.61
56	6.38	7.41	7.95	0	8.86	5.48	5.71	6.5	0	1	0	1	0	0	1	0	0	0	57.37	58.78
57	0	0	12.4	0	0	5.38	5.08	8.77	0	0	0	1	0	0	0	1	1	0	77.59	78.04
58	0	0	0	9.99	0	6.78	0	0	0	0	0	0	1	0	0	0	0	0	88.67	89.47
59	0	0	0	0	0	0	13.08	9.55	0	0	0	0	0	0	0	1	1	0	92.36	89.64
60	0	0	12.15	6.94	9.58	0	0	0	4.88	0	0	1	0	0	0	0	0	0	63.86	68.38
61	0	0	0	8.31	0	8	5.07	0	0	0	0	0	0	0	0	1	0	0	81.13	82.46
62	6.55	8.81	0	0	0	5.74	0	5.54	8.37	0	0	0	0	0	1	0	1	0	67.49	69.72
63	0	2.38	5.45	0	0	3.96	10.39	0	8.17	0	0	0	0	0	1	0	0	0	74.34	74.55
64	0	3.6	6.35	10.15	0	5.01	0	8.69	8.18	0	0	1	1	0	0	0	1	1	71.4	69.61
65	8.55	0	10.23	0	2.72	6.64	0	13.19	0	0	0	0	0	1	1	0	0	0	71.42	73.48
66	0	0	8.98	0	9.33	0	11.69	3.14	0	0	0	1	0	1	0	0	1	0	76.21	76.66
67	0	0	0	0	8.74	5.04	0	0	10.08	0	0	0	0	1	0	0	0	1	81.19	79.82
68	13.13	12.47	9.02	12.86	5.8	3.78	12.31	0	0	0	1	1	1	0	1	0	0	0	50.72	60.81
69	4.23	0	11.98	4.38	13	11.46	10.67	13.95	0	0	0	0	0	0	0	1	0	0	48.82	54.84
70	3.28	10.21	8.42	6.43	6.95	9.05	10.59	0	2.8	1	1	1	1	0	0	0	0	1	46.39	50.6
71	9.71	8.46	8.14	9.32	7.92	4.57	4.93	8.87	11.27	0	1	0	0	1	0	1	1	0	45.33	51.12
72	0	11.41	12.33	5.93	4.99	11.72	10.56	0	10.53	0	0	0	0	0	0	0	0	1	51.71	56.13
73	8.83	0	12.66	13.16	10.16	0	0	0	6.49	0	0	1	1	1	0	0	0	0	60.44	66.18
74	5.73	11.83	6.02	4.32	3.32	13.55	12.38	12.23	2.95	0	0	1	1	0	0	0	0	0	41.91	47.46
75	0	0	11.04	0	0	0	0	8.49	0	0	0	1	0	0	0	0	1	0	85.02	84.06
76	2.58	10.17	11.58	0	13.58	7.33	11.97	4.96	2.6	1	1	0	0	0	1	1	0	1	46.27	52.75
77	8.01	7.62	0	3.79	11.84	5.86	0	0	2.18	1	0	0	1	0	1	0	0	0	59.79	66.31
78	8.26	10.39	0	10.09	12.9	10.22	4.54	5.81	5.21	1	1	0	0	1	0	0	0	0	50.89	53.16
79	0	6.09	0	12.97	12.05	2.71	4.98	8.11	3.07	0	1	0	0	0	1	0	0	1	56.04	59.01
80	5.3	0	0	0	7.74	0	0	11.99	4.45	1	0	0	0	1	0	0	0	0	73.78	74.97

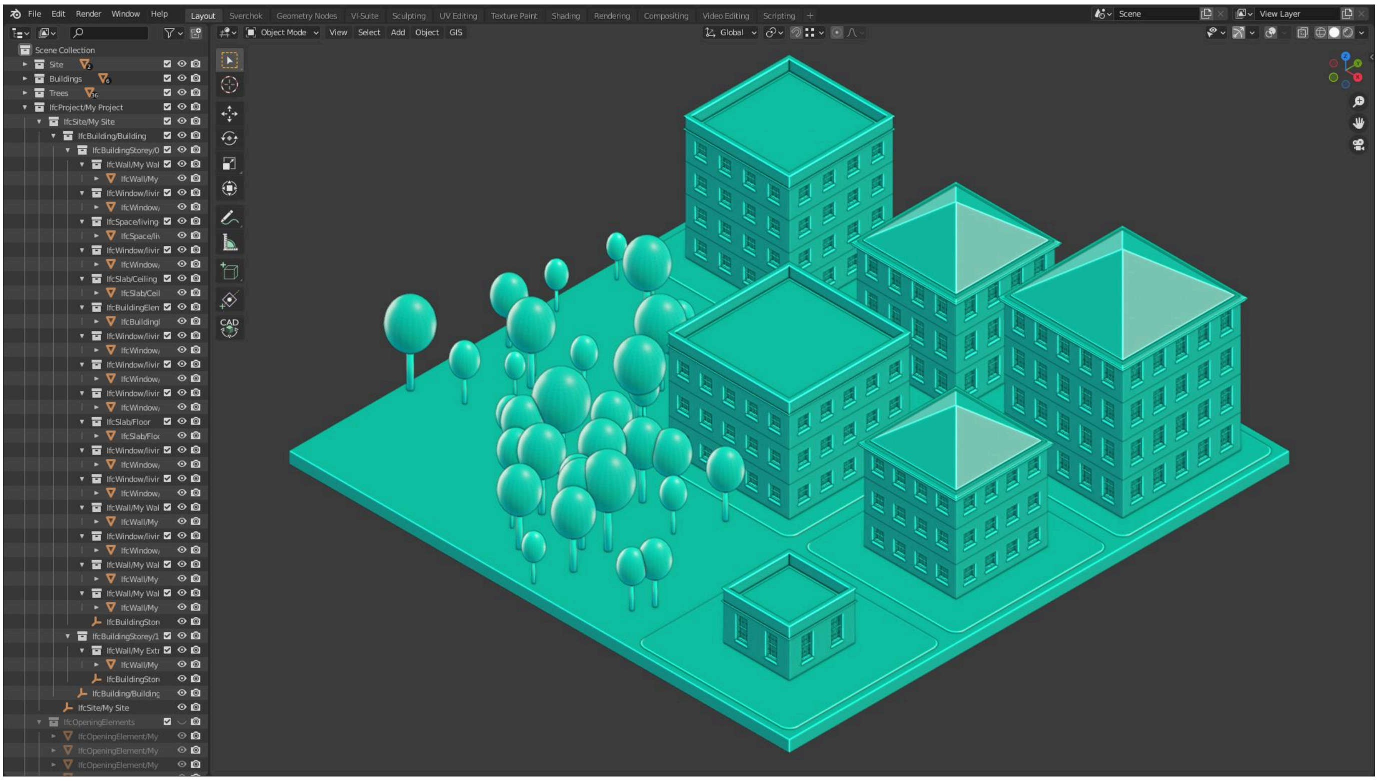
```python
#!/usr/bin/env python
# coding: utf-8

# In[25]:


import pandas as pd
import pickle

filename = '/home/sinasta/Documents/Jupyter/trained_model.sav'
model = pickle.load(open(filename, 'rb'))

a = float(input('Value for Sunlit%: '))
b = float(input('Value for Skyviewfactor%: '))
data_in = [[a,b]]
pred = model.predict(data_in)
m_prediction = pd.DataFrame(pred)
m_prediction.to_csv('/home/sinasta/Documents/Jupyter/results/Situation_prediction.csv',
index = False, header=None)
```

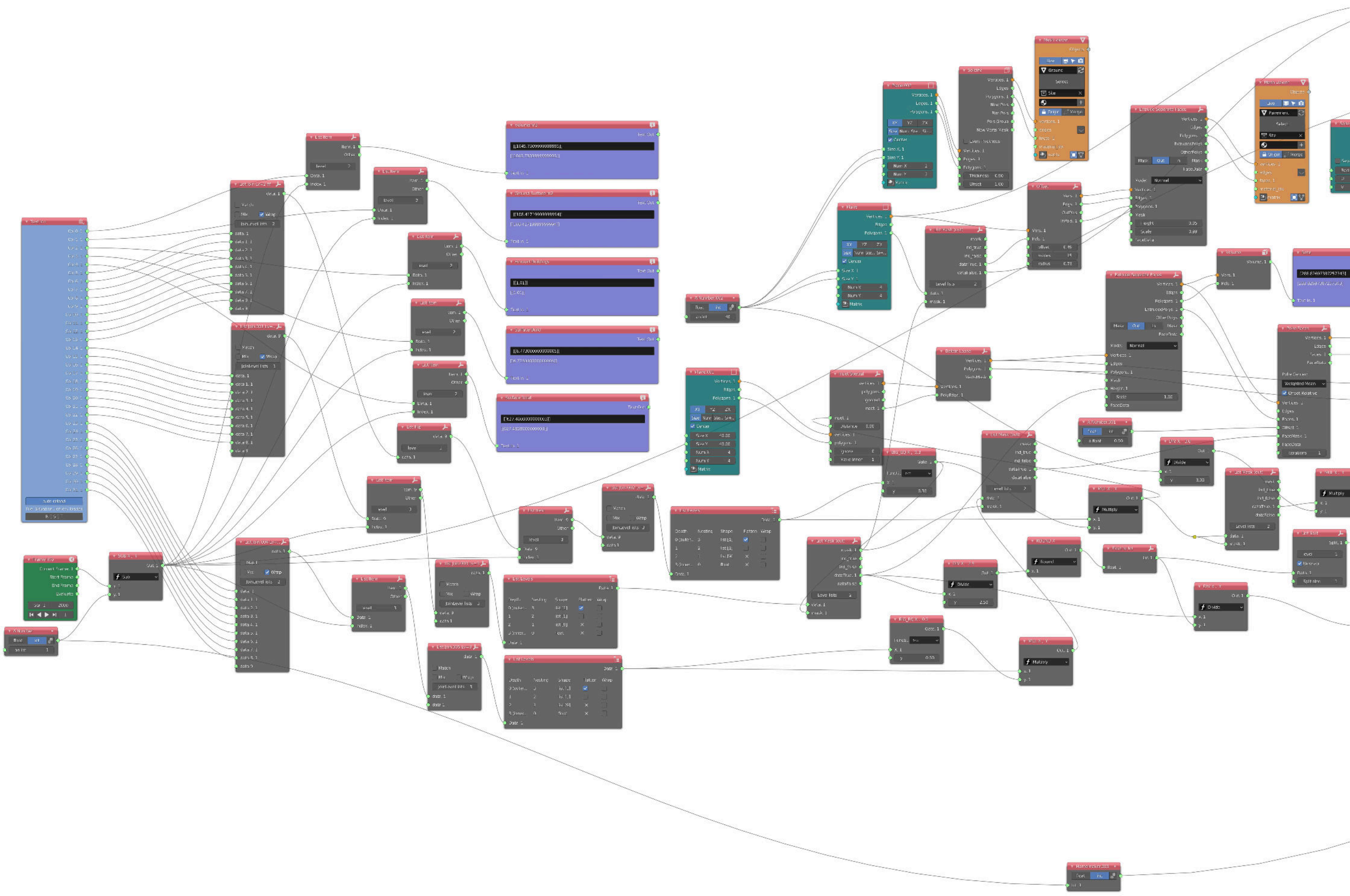
STEP 5 : RESULTS

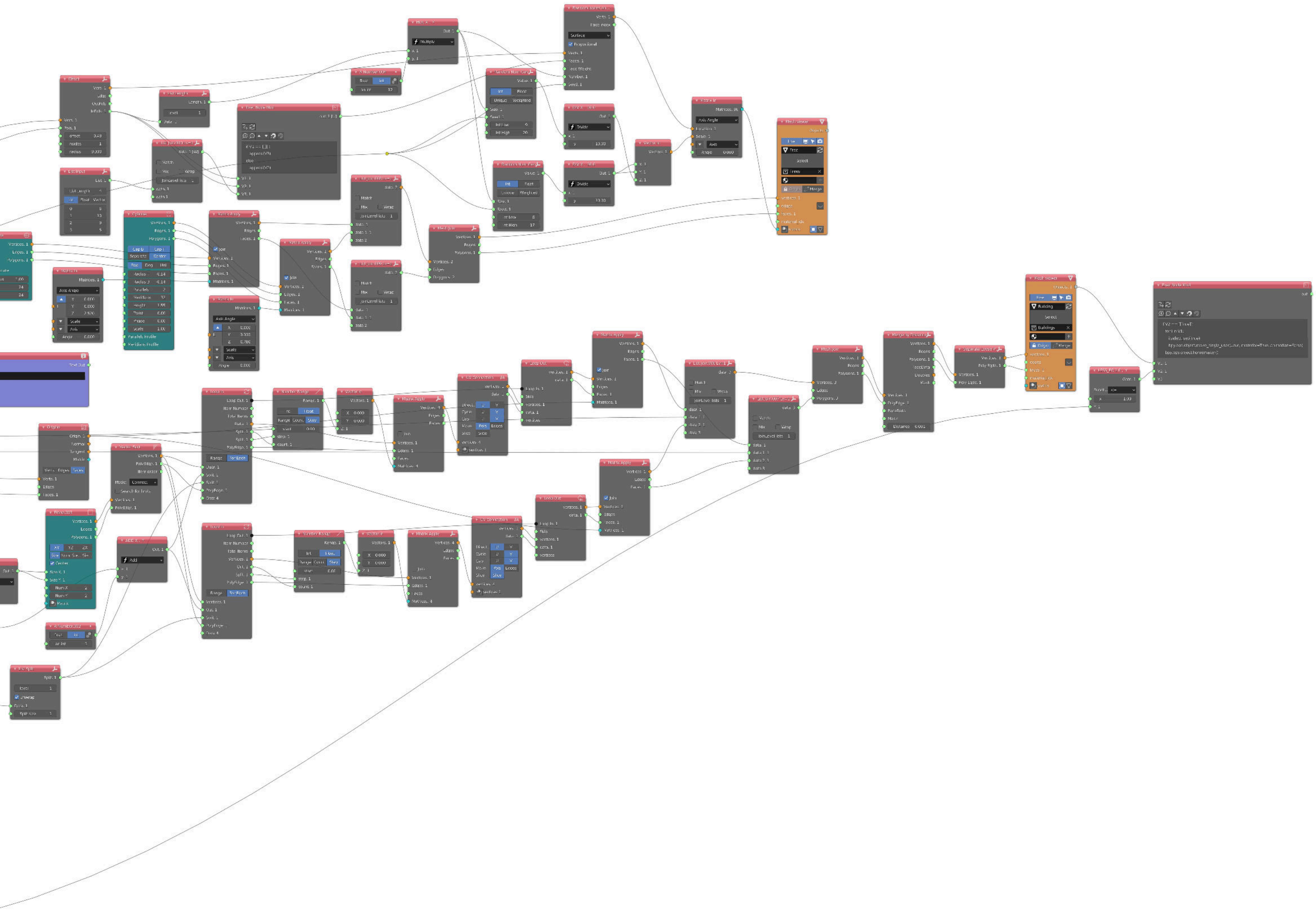
La simulation est la première étape avec une approche pratique plutôt que théorique. Elle consiste à imaginer un contexte représentatif afin de tester les perormances du système développé. Grâce à l'environnement modélisé, une simulation des différentes situations est possible. Il est ainsi possible de faire varier la course du soleil, de changer l'orientation, les dimensions des modules ou l'inclinaison du plan de filtrage.

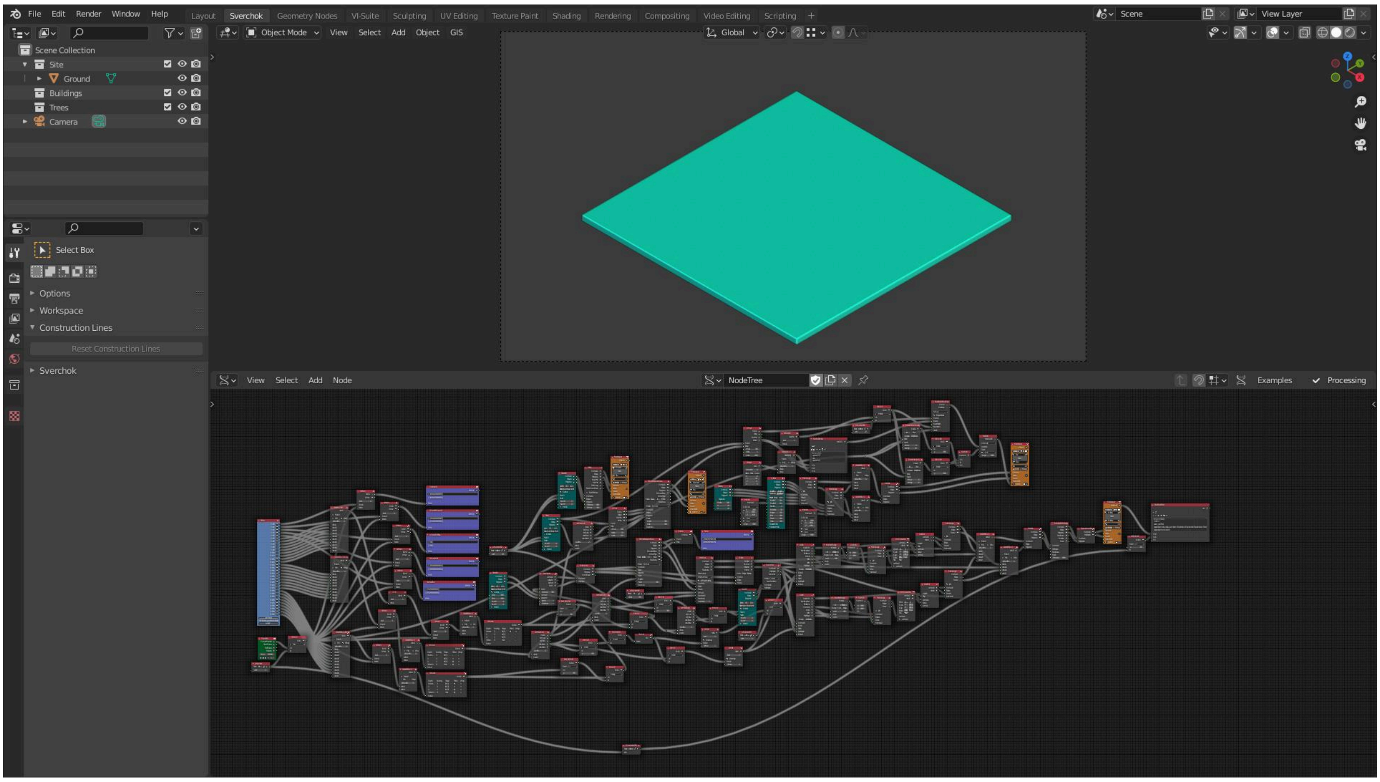
L'utilité de ces simulations est de vérifier les impacts bénéfiques ou non d'une application dans un contexte réaliste.

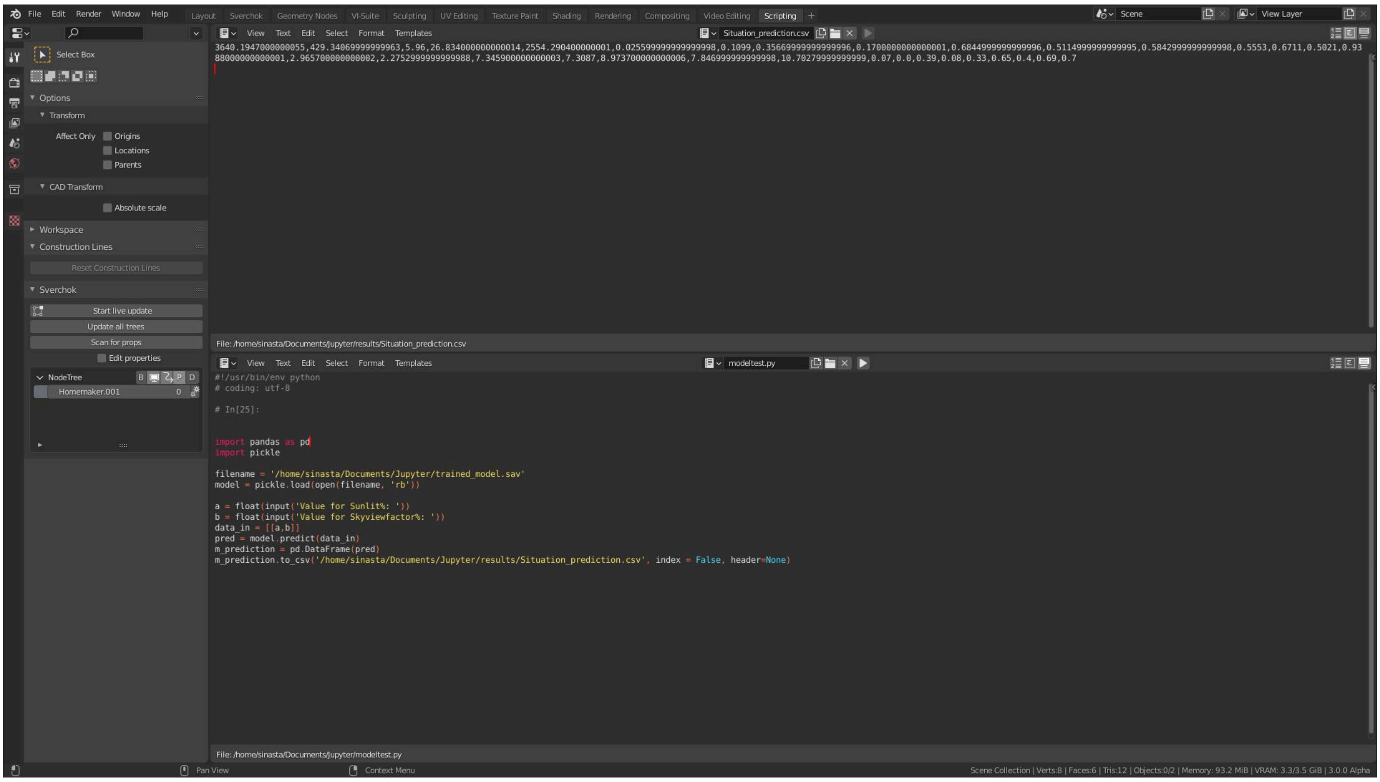

File
Edit
Render
Window
Help
Layout
Sverchok
Geometry Nodes
VI-Suite
Sculpting
UV Editing
Texture Paint
Shading
Rendering
Compositing
Video Editing
Scripting
Scene
View Layer
Object Mode
View
Select
Add
Object
GIS
Global
Scene Collection
Site
Buildings
Trees
Camera
Select Box
Options
Workspace
Construction Lines
Sverchok
Start live update
Update all trees
Scan for props
Edit properties
NodeTree
Homemaker.001
CAD


File
Edit
Render
Window
Help
Layout
Sverchok
Geometry Nodes
VI-Suite
Sculpting
UV Editing
Texture Paint
Shading
Rendering
Compositing
Video Editing
Scripting
Scene
View Layer
Object Mode
View
Select
Add
Object
GIS
Global
Scene Collection
Site
Buildings
Trees
IfcProject/My Project
IfcSite/My Site
IfcBuilding/Building
CAD

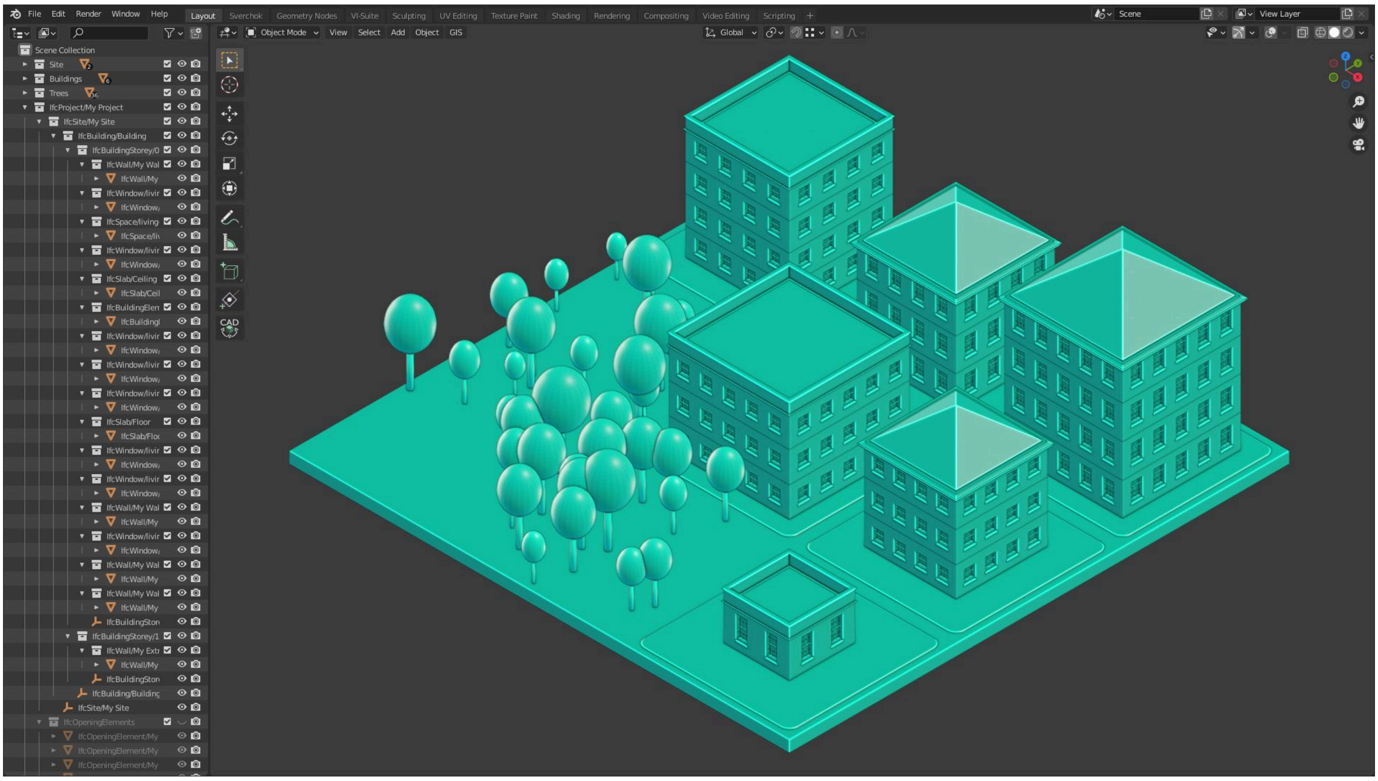







File: /home/sinasta/Documents/Jupyter/results/Situation_prediction.csv
modeltest.py
#!/usr/bin/env python
# coding: utf-8
# In[25]:
import pandas as pd
import pickle
filename = '/home/sinasta/Documents/Jupyter/trained_model.sav'
model = pickle.load(open(filename, 'rb'))
a = float(input('Value for Sunlit%: '))
b = float(input('Value for Skyviewfactor%: '))
data_in = [[a,b]]
pred = model.predict(data_in)
m_prediction = pd.DataFrame(pred)
m_prediction.to_csv('/home/sinasta/Documents/Jupyter/results/Situation_prediction.csv', index = False, header=None)
File: /home/sinasta/Documents/Jupyter/modeltest.py


Terminal
sinasta@sinasta Linux 5.10.32-1-MANJARO x86_64 21.0.2 Ornara
~/Documents/Jupyter >>> python modeltest.py
Value for Sunlit%: 76
Value for Skyviewfactor%:



CONCLUSION

Au cours des quatre dernières décennies, le domaine de l'architecture et d'ingénierie a connu une vaste expérimentation sur le sujet des structures non statiques s'adaptant aux paramètres contextuels, attirant l'attention principalement pour leur caractère artistique et esthétique, la fonctionnalité de ces structures tombant souvent dans l'oubli. Il est clair que l'architecture est en partie un art à apprécier d'un point de vue artistique, mais c'est aussi une science concrète. Les façades cinétiques sont un bon exemple pour symboliser cette ambiguïté.

Les résultats de la simulation montrent qu'une application possible de ce travail théorique pourrait apporter des avantages en matière de confort des habitants et de consommation d'énergie. Comme mentionné dans la première partie, nous vivons un tournant décisif qui va déterminer la responsabilité que nous prenons envers notre planète. Le changement climatique nous oblige à repenser nos habitudes que nous avons adaptées depuis des générations. Le secteur de la construction est l'un des plus grands consommateurs d'énergie primaire et, contrairement à d'autres domaines, il n'est pas très compliqué d'adapter de nouvelles stratégies éco-responsables (passives). Il est donc important pour les architectes et les ingénieurs de s'assurer que leurs projets répondent aux exigences esthétiques et écologiques des années à venir.
Avec un climat à tendance à se réchauffer de plus en plus rapidement et avec des heures d'ensoleillement qui augmentent d'année en année, l'architecte est obligé d'adapter ses projets en concidérant ces évolutions. Les façades cinétiques sont un moyen intéressant de profiter des avantages énergétiques de ce developpement. En optimisant l'utilisation de l'énergie solaire gratuite, nos maisons peuvent atteindre un statut d'autonomie (*nZEB*) qui pourrait avoir des effets de ralentissement concernant la crise écologique.
Il est évident que le mécanisme tel que décrit dans ce travail n'apportera certainement pas les bénéfices démontrés dans la dernière étape. Dans ce travail, des facteurs importants tels que l'énergie grise investie au moment de la production et de la mise en œuvre ne sont pas pris en compte. De même, les coûts d'entretien régulier ne seront certainement pas économiques. Cependant, l'utilisation d'un filtre adaptatif peut apporter des avantages importants dans la construction et des façades cinétiques bien conçues sont bien plus que de simples éléments artistiques.

FOLLOW-UP

L'étude sur les façades responsives traitée dans cet ouvrage est présente sous une forme multidisciplinaire. Chaque étape implique un investissement dans des domaines variés tels que l'ingénierie, la biologie, la chimie, la physique ou les sciences de l'environnement. Ce sont des sciences qui me sont relativement éloignées et il a fallu un effort important pour pouvoir s'approprier les règles de base de ces disciplines. Pour la suite du travail, il est d'interêt de continuer à travailler dans les domaines proches de l'architecture pour pouvoir améliorer les résultats. Ainsi les recherches sur les mécanismes des modules pourraient être faites en tenant compte des propriétés physico-chimiques des matériaux ainsi que de leur résistance qui peuvent être simulées dans des logiciels dédiés à ces fins. En ce qui concerne les pièces de déclenchement, une simulation des champs électro-magnétiques ou des simulateurs de comportement des liquides pourrait aider à concrétiser les concepts. De plus, l'expérimentation cybernétique autour des circuits Arduino peut être poussée plus loin pour mieux comprendre l'intégration et l'interaction de plusieurs capteurs dans un système motorisé. Enfin, la simulation de l'environnement dans la dernière étape est très abstraite et laisse de côté de nombreux paramètres. Ici, il peut être intéressant d'analyser les effets du vent sur la façade ainsi que les propriétés stabilisatrices de la structure. Les analyses de l'énergie et de la lumière du soleil peuvent être menées plus loin pour donner des résultats plus concrets à appliquer à des projets réels. En effet, une meilleure compréhension des logiciels énergétiques tels que *EnergyPlus* aurait permis d'analyser les effets plus en détail.

USED SOFTWARE
(open-source)

MODELISATION 3D

FreeCAD [modelisation parametrique] https://github.com/FreeCAD
Blender [3D creation suite] https://github.com/blender
Openscad [solid 3D cad modeller] https://github.com/openscad
Meshgen [mesh generation tool] https://github.com/jtsiomb/meshgen
Meshlab [mesh processing system] https://github.com/cnr-isti-vclab/meshlab
BRL-CAD [Solid Modeller] https://github.com/BRL-CAD
OpenCascade [3D modeling lbrary] https://github.com/tpaviot/oce
Antimony [modelisateur paramétrique] https://github.com/mkeeter/antimony
Slic3r [toolpath generator] https://github.com/slic3r/Slic3r

ADDONS

Archipak [modelisation architecturale] https://github.com/s-leger/archipack
BenderBim [architecture ifc library] https://github.com/IfcOpenShell/IfcOpenShell
Sorcar [procedural modeling] https://github.com/aachman98/Sorcar
Sverchok [parametric tool] https://github.com/nortikin/sverchok
Vi-Suite [environement analysis] https://github.com/rgsouthall/vi-suite06
A2plus [assembly workbench] https://github.com/kbwbe/A2plus
Assembly4 [Assembly workbench] https://github.com/Zolko-123/FreeCAD_Assembly4
BIM [BIM Workbench] https://github.com/yorikvanhavre/BIM_Workbench
CfdOF [Computational Fluid Dynamics] https://github.com/jaheyns/CfdOF
DesignSPHysics [DualSPHysics fluid simulator] https://github.com/DualSPHysics/DesignSPHysics
Fasteners [Fasteners workbench] https://github.com/shaise/FreeCAD_FastenersWB
FcGear [Gear workbench] https://github.com/looooo/freecad.gears
Slic3r-tools [slic3r workbench] https://github.com/limikael/freecad-slic3r-tools
TechDraw [2D CAD workbench] https://github.com/WandererFan/FreeCAD-TechDraw

FEM [Finite element analyse workbench] https://github.com/FreeCAD/FreeCAD/tree/master/src/Mod/Fem

VISUALISATION

LuxRender [Render engine] https://github.com/LuxCoreRender
Cycles [Render engine] https://github.com/boberfly/cycles
EEVEE [Render engine] https://github.com/sobotka/blender
Appleseed [Render engine] https://github.com/appleseedhq/appleseed
ParaView [Data Analysis and Visualization Application] https://github.com/Kitware/ParaView

SIMULATION

Radiance [Lighting simulation tool] https://github.com/NREL/Radiance
OpenStreetMap [Map library] https://github.com/openstreetmap
CodeAster [Structure and Thermomechanics analysis] https://github.com/ralic/Code_Aster
CodeSaturne [turbulence analysis] https://github.com/code-saturne/code_saturne
EnergyPlus [building energy simulation program] https://github.com/NREL/EnergyPlus
GMSH [finite element mesh generator] https://gitlab.onelab.info/gmsh/gmsh
OpenFOAM [computational fluid dynamics software] https://github.com/OpenFOAM
Ladybug-Tools [environmental simulation] https://github.com/ladybug-tools

MANIPULATION 2D

Qcad [2D CAD] https://github.com/qcad/qcad
LibreCAD [2D CAD] https://github.com/LibreCAD/LibreCAD
Inkscape [vector graphics software] https://github.com/inkscape/inkscape
GIMP [Image Manipulation Program] https://github.com/GNOME/gimp
Scribus [Desktop publishing software] https://github.com/scribusproject/scribus

GeoGebra [Math visualizer] https://github.com/geogebra/geogebra
Gthumb [Image viewer and editor] https://github.com/GNOME/gthumb
Solvespace [Parametric 2D/3D modeler] https://github.com/solvespace/solvespace
LibreOffice [office Suite] https://github.com/LibreOffice/core
PDF Tricks [PDF Manipulator] https://github.com/muriloventuroso/pdftricks
Evince [PDF Viewer] https://github.com/GNOME/evince
Gnome-LaTeX [LaTeX editor] https://github.com/GNOME/gnome-latex

AUTRES

FFmpeg [Audio and video manipulator] https://github.com/ffmpeg/ffmpeg
Arduino IDE [Developement environement] https://github.com/arduino/arduino-pro-ide
Zotero [research organizer] https://github.com/zotero/zotero
Ghostscript [postscript language interpretor] http://git.ghostscript.com/
DeepL [Translator] https://github.com/vsetka/deepl-translator
Impress [Presentation framework] https://github.com/impress/impress.js/

BIBLIO-
GRAPHY

MONOGRAPHIES

HUANG, W. ZHENG , H. 2018. Architectural Drawings Recognition and Generation through Machine Learning, Mexico City, ACADIA

CHAILLOU, S. 2019. AI + Architecture towards a New Approach, Massachusetts, Harvard GSD

TAMKE, M. et al. 2018. Machine learning for architectural design: Practices and infrastructure, Cottbus, IJAC

FAISAL, A. 2020. The impact of AI in architectural design, Abu Dhabi, Abu Dhabi University

EXEMPLES

Evolving Floorplans
Consultable: https://www.joelsimon.net/evo_floorplans.html [disponible le 12 février 2021].

A first step toward the future of neighborhood design
Consultable: https://www.sidewalklabs.com/blog/a-first-step-toward-the-future-of-neighborhood-design/ [disponible le 12 février 2021].

MX3D Bridge
Consultable: https://mx3d.com/projects/mx3d-bridge/ [disponible le 12 février 2021].

AI & Architecture
Consultable: http://stanislaschaillou.com/articles.html [disponible le 12 février 2021].

Ai in Architecture
Consultable: https://www.youtube.com/watch?v=PXlsjk5nKyc [disponible le 12 février 2021].

Generative design tool for architects
Consultable: https://finch3d.com/ [disponible le 12 février 2021].

Topologic
Consultable: https://topologic.app/ https://twitter.com/topologicBIM [disponible le 12 février 2021].

Talk on Topologic
Consultable: https://community.osarch.org/discussion/131/talk-on-topologic
[disponible le 12 février 2021].

Pattern Languages Homemaker Evolutionary Design
Consultable: https://twitter.com/brunopostle [disponible le 12 février 2021].

Topologise Blender Add-On
Consultable: https://www.youtube.com/watch?t=279&v=bM4ewlvLJ98&feature=youtu.be
[disponible le 12 février 2021].

TopologicPy
Consultable: https://github.com/wassimj/topologicPy https://github.com/wassimj/topologicsverchok
[disponible le 12 février 2021].

Homemaker Add-On
Consultable: https://github.com/brunopostle/homemaker-addon [disponible le 12 février 2021].

MÉCANISMES

STAVRIC, M. et al. 2016. Parametrics of Movable Polyhedral Models in Performative Architecture, Graz, Graz University of Technology

AL-KHALFA, D. et al. 2016. Prototyping for kinetic architecture, Damascus, Damascus University

ABDULMAJID, K. et al. 2015 « Innovations in dynamic architecture », Journal of Facade Design and Engineering, https://content.iospress.com/articles/journal-of-facade-design-and-engineering/fde0040

The Spaces
Consultable: https://thespaces.com/kinetic-architecture-hong-kong/ [disponible le 13 juin 2020].

Strandbeest
Consultable: https://www.strandbeest.com/ [disponible le 13 juin 2020].

Bimetals
Consultable: http://www.iaacblog.com/programs/thermatrix_bimetals-phase-2/ [disponible le 13 juin 2020].

Islamic Patterns
Consultable: https://www.geno-morph.com/igp [disponible le 13 juin 2020].

FONCTIONNEMENT

KARANOUH, A. et al. 2017. Innovations in dynamic architecture, Bath, Journal of Facade Design and Engineering

SHAWUL, H. et al. 2018. Building Envelop Design Lab, Milano, Politecnico Milano

ZARZYCKI, A. 2014. Reflections on Computational Design Through Interactions With Materiality and Physical Mock-Ups, New Jersey, New Jersey Institute of Technology

LAAB
Consultable: https://www.laab.pro/?portfolio_page=harbour-kiosk [disponible le 13 juin 2020].

Responsive Structures
Consultable: https://www.geno-morph.com/rg [disponible le 13 juin 2020].

ENVIRONEMENT

GREIVING, S. et al. 2013. Climate Change and Territorial Effects on Regions and Local Economies, Dortmund, TU Dortmund

POTIENKO, N. et al. 2018. The Global Experience of Deployment of Energy-Efficient Technologies in High-Rise Construction, Samara, Samara State Technical University

GIEDROWICZ, M. 2015. Parametric Design for Ecological Purposes – Case Studies and Algorithm Examples, Poznan, Poznan University of Technology

ELGHAZI, Y. et al. 2014. Daylighting Driven Design: Optimizing Kaleidocycle façade for hot arid climate, Cairo, AASTMT

SIMULATION

PANYA, D. et al. 2019. PA Methodology of Interactive Motion Facades Design through Parametric Strategies, Kyungpook, Kyungpook National University

SUJI, C. et al. 2015. Calculation Method of Shaded Fraction According to the Movements Type for Kinetic Facade, Inha, Inha University
Dynamic Daylight
Consultable: https://drajmarsh.bitbucket.io/daylight-box.html [disponible le 13 juin 2020].

Shading Box
Consultable: http://andrewmarsh.com/apps/staging/shading-box.html [disponible le 13 juin 2020].

Heat Balance
Consultable: https://drajmarsh.bitbucket.io/heat-balance.html [disponible le 13 juin 2020].

Psychrometric Chart
Consultable: https://drajmarsh.bitbucket.io/psychro-chart2d.html [disponible le 13 juin 2020].

2D Sun-Path
Consultable: https://drajmarsh.bitbucket.io/sunpath2d.html [disponible le 13 juin 2020].

Weather Data
Consultable: https://drajmarsh.bitbucket.io/weather-data.html [disponible le 13 juin 2020].

Arduino Circuit
Consultable: https://www.circuito.io/ [disponible le 13 juin 2020].