Situated Toolmaking for Creative Computing:

A Framework for Context-aware Machine Learning in Design, Art, and Making

Ardavan Bidgoli

Doctoral Candidate, School of Architecture, Carnegie Mellon University

Committee:

Dr. Daniel Cardoso Llach (Chair)

Associate Professor, School of Architecture, Carnegie Mellon University

Dr. Eunsu Kang

Visiting Professor of Art and Machine Learning, Carnegie Mellon University

Prof. Golan Levin

Associate Professor of Art, School of Art, Carnegie Mellon University

Dr. Barnabás Póczos

Associate Professor in Machine Learning, Machine Learning Department, Carnegie Mellon University

Fall 2019

Table of Contents

Table of figures

Table of Tables

Introduction

Recent advancements in artificial intelligence (AI) and its sub-branch machine learning (ML) have led a growing number of artists, designers, and architects to explore these techniques' affordances in developing tools that support their creative practices such as designing and making. Despite a wave of exciting results, there is a significant gap in the current state of ML research when it comes to grounding ML tools in their specific contexts. While a lion-share of efforts in the machine learning research community is focused on developing novel algorithms and improving their efficiency (Simard et al., 2017), fewer resources are dedicated to informing the design process with inputs from end-users as well as from the context in which these models will be eventually deployed.

In the current state of applied machine learning, the domain experts' or the end-users' engagement in the toolmaking process is mostly limited to providing data, domain-related information, and feedback on the final products. Moreover, machine learning experts—who might not have a strong grasp of that specific practice—are responsible for tasks, including but not limited to, curating data sets, developing algorithms, training and fine-tuning models, evaluating models' performance, and deploying solutions (Amershi, Cakmak, Knox, & Kulesza, 2014, p. 105). This distinct separation between expert users and ML experts intensifies the prevalent abstraction as well as the decontextualization problem which have been critical challenges of AI.[1] This problematic approach results in tools that are incapable of serving in the real environments, interrupting the expert users' workflow, or introducing unexpected hurdles in the process.[2] My research aims to alleviate these issues by developing a framework that accommodates expert users and the contextual data as integral elements of the toolmaking process.

Creative users—i.e., designers, artists, and architects—employ machine learning models to create tools that can support their creative activities in design and making.[3] These activities are usually categorized under two main branches: "creative computing" and "computational creativity". Creative Computing—the primary subject of this research—is the intersection of creative practices and computing. It refers to the process of using computing in creative activities. This approach stands in contrast with Computational Creativity, which stands for the use of computational methods to mimic human creative actions (Fiebrink, 2019).

In creative computing toolmaking using current ML techniques—the process of developing computational tools for creative activities such as architecture, art, design, and making—the involvement of ML experts in the workflow is inevitable. This stems from the fact that the expert users in creative domains are rarely

---

[1] The decontextualization problem that is thoroughly addressed in (Forsythe, 2001) will be discussed under The Situatedness Gap section.

[2] An interesting track of discussion on the incompleteness of representation can be traced in Yann LeCun's concerns on the limitations of deep reinforcement learning.

[3] A growing number of artists are exploring the new boundaries of ML and Art. Gene Kogan, Memo Akten, Mario Klingemann, Kyle McDonald, Anna Ridler, Studio for Creative Inquiry, Obvious team, Runway team, and Magenta team at Google are just a few examples of a thriving community of ML artists who use state-of-the-art ML tools to serve as a means of creative expression. I discussed a few examples of their works previously (Bidgoli, Kang, & Llach, 2019).

in possession of technical skills required to design, train, and deploy complex machine learning-based tools.[4]

This research proposes a paradigm shift in the process of ML-based creative computing toolmaking that grants expert users more involvement and control over the process, thus mitigating the adverse effects of the decontextualization. It proposes a simplified and accessible approach for domain experts with limited knowledge of computer programming to train and deploy ML models. The proposed approach can potentially improve expert users' ability to build their own creative computing tools and introduce contextual data into it without engaging with the complexities of the backend ML systems.

My hypothesis is that employing human-centered machine learning techniques such as interactive Machine Learning (iML), Learning from Demonstration (LfD), and generative models can help expert users to achieve these goals:

1) Enhancing their ability to interface and interact with a learning system;

2)Integrate social and physical registers of their creative practices—i.e., personal preferences, subjective assessments, material behavior, tool characteristics—to the toolmaking process and build learning systems that support situated approach to creative computing toolmaking;

3) Utilizing such situated tools towards generating new instances of their practice.

The expected outcomes of this research constitute of 1) a theoretical framework that explains the situated approach to ML-based creative computing toolmaking, 2) an implementation of the framework in the format of a toolkit, and 3) a series of user studies to study the framework and corresponding toolkit in practice.[5]

Motivation, Importance

The motivation behind this research is multifaceted. Most notably, the potential value that it offers to creative users—i.e., designers and architects. It proposes a framework for accessible and situated ML toolmaking.[6] This research helps the creative users to develop creative computing tools without falling in the common pitfalls of computational design tools such as inflexibility and reduction.[7]

---

[4] It is worth mentioning that the community of ML artists is continually growing. ML artists are actively engaging with new audiences by running workshops across the globe. Several universities include ML and Art in their curriculum and various online programs are available for artists to get hands-on experience with ML. Also, companies such as Google are nourishing the creative application of ML as a testbed for their research projects.

[5] In these user studies, a group of expert users from various creative fields, including but not limited to, building trade, design, and art will utilize the toolkit to develop their creative computing tools and apply them in practice. The user studies will be discussed in more details in the User Studies section.

[6] While not being a primary contribution of this proposal, this framework can potentially help ML experts to engage in creative activities using their custom-made tools as well.

[7] Daniel Davis has discussed some of these challenges in his Ph.D. dissertation. He observes that there is a limited documentation of parametric modeling—as one of the computational design approaches—challenges and shortcoming in the literature of computational design. He then proceeds to list some of these challenges based on Rick Smith's technical note. Smith provides us with five critiques of parametric modeling: 1) necessity of making many decisions upfront, or as he calls it *upfronting*, 2) necessity of anticipating future design changes, 3) incapability of dealing with major design changes, 4) challenge of foreseeing the effects of changes in one part of the model on the other parts, and 5) limitations on sharing and usability due to over-complicated models than cannot be comprehended by anybody other than the original designers (Smith 2007 as cited in Davis 2013). Stasiuk and Thomson suggest that the first three critiques are referring to the necessity of predetermination in the design and its features (Stasiuk & Thomsen, 2014).

Another motivation is the framework's focus on context—i.e., the social, physical, and spatial context—as well as the emphasis on human-machine bounds. This approach advocates the point of view that defines ML tools in relation to their makers, users, and the specialties of their context. A distinct point of view offered herein is that it considers the toolmaker, learning system, and the context as "mutually constitutive"[8] to address the shortcomings of current ML efforts in design and architecture. I expect that raising attention to these factors contributes to the counterpoint discussions on the conception of ML tools as being "autonomous" and "stand-alone agents".

Exploring new forms in which toolmakers interface with their toolmaking workflows is another driving force behind this research. This research advocates the use of data as a malleable design material and the main means of interaction between the expert users and the ML model (Cardoso Llach, 2017). Users moderate the process of selecting or generating training samples, decide on the specific parameters to be included in each sample, and sculpt custom data sets to model a specific concept. It is upon the user to thoughtfully decide on the inclusion of contextual, subjective, and personal inputs alongside the datapoints that are typically included in ML data sets. Thus, data as an interface approach can facilitate the incorporation of physical and social elements of a practice in the tool-making process.

Subsequently, this research paves the way to posit ML-based tools with respect to the context and human agents—toolmakers, users, and audiences—rather than stand-alone and autonomous agents. The inclusion of human agents in the process of making ML-based tools is gaining attention in the shadow of ongoing debates on the future of work in the presence of "autonomous" systems.[9] For example, a construction worker can interface and teach a robotic agent to work in tandem with him/her through demonstration to deliver material or to utilize tools in a specific way that matches the human agent's workflow.[10]

The other driving force behind this research is to explore the unexpected interactions between the proposed toolmaking framework and the toolmaker throughout various phases of the tool's life-cycle—the toolmaking process as well as runtime.[11] The particular purpose of this research is to study such dynamic interactions. For example, the unexpected event of an outlier sample in the training data set, a surprising generated sample, or a never-seen-before design space, requires the toolmaker's intervention to steer the learning process or to provide new learning samples. These interactions help the toolmaker to iteratively shape the tool by following its preferences and simultaneously build a personal perception of the tool and its properties.

[8] This term is loosely borrowed from Lave and Wenger (1991).

[9] It is commonly believed that with the introduction of autonomous machines in manufacturing, these machines will take over certain jobs while the current human labor force will shift to occupy newly created jobs, i.e., training, maintaining, and supervising these autonomous machines. However, most of the jobs that these machines will takeover are the ones that require less training and experience. In most cases, workers in these jobs are the most vulnerable ones in the job market with the lowest wages. It is naïve to assume that these workers can smoothly prepare themselves for new complex responsibilities that autonomous systems entail. These new jobs usually demand for educated workers with longer training period and more experience, which is a different demographic section than those who are losing their jobs. The approach that I am proposing aims to reduce the gap between the skills required for the jobs lost and the jobs created as these autonomous agents enters the job market.

[10] This approach acknowledges the co-existence of human agents as well as autonomous agents on the job sites, rather than total replacements of human agents. It also advocates the continual participation of existing workforce in the process of teaching these agents.

[11] This is partially informed by the studies of Cardoso et al. in the field of architectural robotics and the interactions between a designer and the tool (Cardoso Llach, Bidgoli, & Darbari, 2017).

An additional motivation of this research is to contribute to the ongoing discussions on bias in machine learning. Machine learning research is highly dependent on the publicly available data sets that are skewed and biased towards specific races, gender, geographic regions, art genres, etc. Generalizing the application of ML models that are trained on these data sets is a source of debate. However, this research explores the positive side of biased data sets. The proposed approach embraces the *generative potentials of bias* that resides in the personal judgments and subjective metrics of each expert user to form a customized tool.

Definitions

Creative Users: Throughout this document, the term creative users refers to individuals or groups who are creating ideas or artifacts in various fields, including but not limited to the fine arts, music, composition, performance, design, architecture, or any other domain where creative engagement is a key element of the practice.[12]

Creative Computing: Creative Computing is the intersection of creative art and computing. It refers to the process of using computing in creative activities.

Machine Learning: Machine Learning is a sub-branch of artificial intelligence that studies algorithms that can accomplish a given task by observing a series of solution examples, rather than explicitly programming the solution. [13] ML algorithms rely on statistical methods to gradually improve their performance through an iterative cycle of experiences. The experience in this context may refer to observing a data set of unlabeled data (unsupervised learning), a data set of labeled data (supervised learning), a series of simulations (reinforcement learning), or a series of demonstrations (learning from demonstration).

Interactive Machine Learning: Interactive machine learning (iML)—first introduced by Fails and Olsen (Fails & Olsen, 2003)—refers to "algorithms that can interact with agents and can optimize their learning behavior through these interactions, where the agents can also be human" (Holzinger, 2016). In iML, the user can iteratively add new learning samples to steer the learning direction until the desired outcome is achieved. Interactive machine learning can help expert users introduce their knowledge and insight into the learning process without having a comprehensive understanding of the machine learning algorithm (Dudley & Kristensson, 2018, p. 8).

Learning from Demonstration: Built on concepts of "*learning from examples*" from the late 1970s, Learning from Demonstration (LfD) is the process of learning a task by observing an expert user (Argall, Chernova, Veloso, & Browning, 2009). It aims to frame the learning task as human-teacher/machine-learner interactions. LfD has been widely used in robotics and proved to be useful when we need to program a robot not by a robotists, but by an end-users (Rahmatizadeh, Abolghasemi, Behal, & Bölöni, 2016).

Generative Models: A generative model (GM) in machine learning refers to a model that can be trained on an unlabeled subset of a distribution p_{data} and learns an estimate representation of that distribution, p_{model}. Users can leverage GMs to generate new instances that resemble the ones in the training set by drawing

---

[12] These definitions are borrowed from Rebecca Fiebrink's, a pioneer of Art and ML and senior lecturer at the Department of Computing at Goldsmiths University, definition of *creative practitioner* and *creative domains*: "people creating ideas or artifacts in a broad set of domains. They include creators in the fine arts, music composition and performance, and theater and performance art, as well as creators of new indie games and "makers" of other hard-to-pigeonhole artifacts and experiences." She also defines creative domains as domains in which creative expression is a primary goal (Fiebrink, 2019, p. 2).

[13] Tom Mitchel defines learning algorithms as "[a] computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E." (Mitchell, 1997).

samples from this distribution. From this point of view, generative models are different from the discriminative models that are mapping features to labels and have been widely used for tasks like image classification.[14]

Latent Space (in Generative Models): In generative models, latent vector refers to the internal layers of the model that encode learned features of the training data. For example, in an autoencoder, the latent space is the bottleneck layer. In GANs, on the other hand, the latent vector is the noise vector(s) that feed(s) the generator. Navigating the latent space refers to the act of replacing the latent vector with different values to explore the outputs of the generative model.

State-of-the-art ML in Creative Practices

The latest boom of Machine Learning in the mid-2010s has raised a new wave of interest among artists and designers to explore the intersection of art and artificial intelligence.[15] In this period, artists are able to easily access large data sets—i.e., Google's Art & Culture project—[16]open-source machine learning models—i.e., Ian Goodfellow's Generative Adversarial Networks—and powerful hardware to train them—most notably GPUs and cloud services. These factors resulted in a wide range of creative inquiries in the application of ML in writing, painting, music composition, game design, choreography, and interactive installation (Chris Donahue, Ian Simon, & Sander Dieleman, 2018; Kyle McDonald, 2018; Mario Klingemann, 2017; Memo Akten, 2017; Radford et al., 2018; Sharp, 2016; J. Zhu, Liapis, Risi, Bidarra, & Youngblood, 2018).

Architects also joined this wave and started applying machine learning to enhance various tasks throughout a project's life cycle, from pre-design studies to occupancy evaluation (As, Pal, & Basu, 2018).[17] Machine learning models can help architects develop design tools without explicitly declaring the geometrical topologies or design's features since these models can recognize these features from a thoughtfully curated data set. This is a point of departure from the existing computational methods where a level of predetermination is essential to make flexible models and reduce the costs associated with design changes. This characteristic is promising as it theoretically renders machine learning design tools less prone to

[14] Some researchers refer to the generative aspects of machine learning models as the *unconventional* application of ML. For example, Fiebrink describes *unconventional* applications, as the use of generative models to "produce new content that is "similar" to the training set" (Fiebrink, 2019). In contrast, she lists applications such as processing, reasoning, prediction, or classification of data as example of *conventional* applications of ML. This research avoids this terminology, to prevent from any confusion in the future.

[15] This new wave should not mislead us about the origins of computational tools in creative activities. We can trace the application of computational tools back to the Postwar era. Since then, the conception of computational tools for design has been thought of in various ways, as the "perfect slave", liberating the designers from the manual labor, as well as "collaborative partner" that creatively contributes to the process of making. In any of these capacities—regardless of the correctness of these two extremes and the underlying anthropomorphism— computational tools are artifacts that are designed and created by human agents (Cardoso Llach, 2015, p. 54) and should be studied in relation to their human makers as well as the context.

[16] https://artsandculture.google.com/

[17] A brief glance at the proceedings of Computer-Aided Design conferences reflects this enthusiasm. A review of articles presented at the most recent eCAADe | SIGraDi conference reveals the steady growth of enthusiasm among the architects to explore affordances of ML in various field of architecture, such as fabrication, generating, recognition, and analysis of drawings, predicting preferences, designing fire egress (Eisenstadt, Langenhan, & Althoff, 2019; Jabi, Chatzivasileiadi, Wardhana, Lannon, & Aish, 2019; Kinugawa & Takizawa, 2019; Newton, 2019; Rossi & Nicholas, 2019; Wei et al., 2019).

oversimplification and excessive abstraction, which are commonly associated with conventional computational design methods (Stasiuk & Thomsen, 2014).[18]

Toolmaking for Creative Computing with Machine Learning

In Merriam-Webster dictionary, *tool* is defined as an apparatus or instrument that is required in the practice of activity or profession.[19] In the context of this research, the word *tool* is used as a broad term referring to creative computing tools, including computational design and making tools. Accordingly, the term *toolmaking* refers to the process of studying, designing, and prototyping a creative computing tool for a specific design or making task. *Toolmaker*, in this context, is narrowed down to the individuals, or the groups of creative users who make creative computing tools for the benefit of their practices or other practitioner.

Toolmaking is an integrated part of the design process. It is informed by the task conception and the solution and simultaneously informs the designer's conception of the task and affects the solution.[20 & 21] This dynamic relationship between the designer, solution, and toolmaking process is a crucial part of the design process and a discontinuity in this cycle can break this productive relationship.[22] However, in the context of machine learning toolmaking for creative activities, two factors obstruct the formation of such relationship: 1) accessibility gap, and 2) situatedness gap.

Accessibility Gap

Using machine learning for creative practices chiefly relies on frameworks that are primarily aimed for researchers and developers with extensive prior exposure to machine learning (Roberts, Hawthorne, & Simon, 2018). Understanding the behavior of tools that are run by ML is essential to toolmakers' ability to control and harness their potentials and minimize their harm (Rahwan et al., 2019, p. 477). While toolmakers are expert in their field of work, they usually have limited knowledge of ML. This affects the

---

[18] For a more comprehensive review of ML in architecture, please refer to Appendix A.

[19] https://www.merriam-webster.com/dictionary/tool

[20] This dynamism of a man and its tool is well reflected in this famous quote allegedly associated to Marshall McLuhan: "*We shape our tools and then our tools shape us*".

[21] To illustrate this point, we can think about the process of making a "bricklaying tool" for brick alignment. The toolmaker may start with a simple yarn, fixed between two anchor points, to align straight walls. But as the design proceeds, the toolmaker faces a new challenge: curved walls. The toolmaker improvises a new setup to utilize the same yarn and anchors. This time the yarn will be fixed only on one side, while the other side freely rotates around the fixed anchor point. This process may have happened the other way around as well. For example, the toolmaker realizes that by fixing only one end of the yarn, it can locate the position of the bricks for a curved wall. Eventually, the toolmaker may notice that by fixing the yarn (or a more reliable substitute, a metal chain) to a pole, it can find the location of bricks to construct a dome. By each rotation of the chain around the pole, the toolmaker locates the position of bricks in one layer. As the chain twists around the pole, its length gradually shrinks. This will help the toolmaker to place the bricks in the new layer slightly closer to the centerline of the dome. Eventually, the chain completely wraps around the pole, indicating the location of the last piece. As we see in this example, the tool making process and the design process can inform each other while enriching the design process.

[22] The discussions on the dynamic relationship between human and tools is gaining more attention in the shadow of the recent AI advancement. Iyad Rahwan, the director of the Center for Humans and Machines at the Max Planck Institute for Human Development, argues that human and machines can inform, mold, and alter each other's behavior in various ways. Humans change the machines behavior by engineer algorithms, actively provide training data, or being observed by data collection machines. On the other hand, machines affect our behavior, social fabric, and the political landscape through their omnipresent role in our routine decision making procedures (Rahwan et al., 2019, p. 483).

toolmakers' ability to integrate their rich toolmaking workflow with ML. Thus, they usually limit themselves to off-the-shelf data sets and pre-trained models.

To address this issue, several researchers and artists have focused on making machine learning design tools more accessible to designers and architects. Most of these attempts address the elimination of technical challenges by simplifying the data pipelines, designing user-friendly interfaces or using platforms that are already popular among artists, providing pre-trained models, and integrating cloud computing services for faster computation.

In 2016, Google Brain team released Magenta, an opensource and free project that includes libraries and code snippets to help artists who work with deep learning models to generate music, images, and drawings (Magenta, 2019). Since then, several artists have used Magenta as the backend library to train their ML-based musical tools—mostly over open source data sets. However, using Magenta requires a moderate background in ML and programming. Thus, it is most suited to serves as the backend for other creative computing tools.

Runway—which was initially started as a thesis project at the New York University—is aiming to address this challenge. It is an ML toolkit with a graphical and user-friendly interface that incorporates several open-source ML models developed either by the core team or the community of users. It allows users select and match models and feed various data formats as input and collect the outputs. It has various plug-ins/APIs to integrate with other popular toolkits among artists, such as Open Framework, Processing, Rhino, and Photoshop (Runway, 2019).[23]

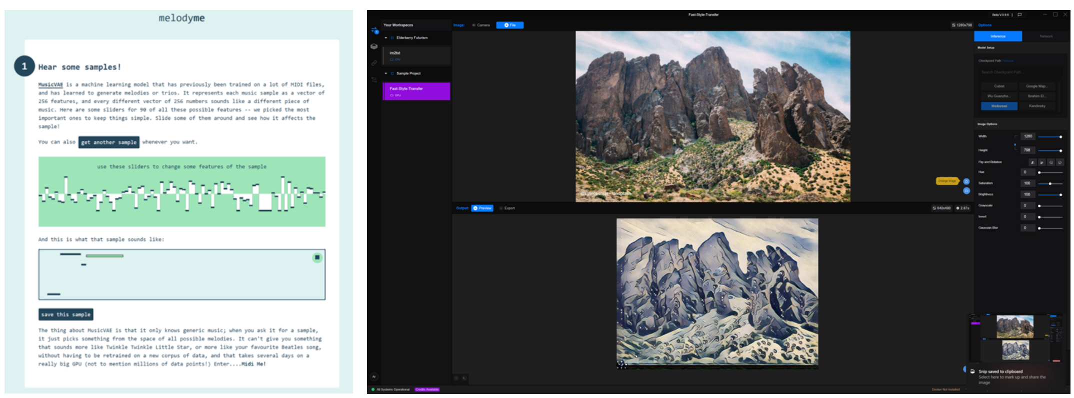

Figure 1- MidiMe (left, image from (Monica Dinculescu, Jesse Engel, & Adam Roberts, 2019)*, Runway interface (right).*

Projects such as Runway ML are focused on the training and inference phase of machine learning-based toolmaking by providing better graphical user interfaces (GUI), pre-trained models, and integration with other platforms. However, these projects generally ignore two important factors: 1) the role of transparency in the way toolmakers and the toolmaking apparatus interface each other; 2) the social and physical ties among the toolmakers, tool, and the design problem.

Regarding the first factor, Runway's graphical user interface and its integrations are a step towards a more transparent and accessible ML for artists. However, its affordances are limited when it comes to the human-

[23] Runway is recently released its public Beta version; the technical details of this version is yet to be explored.

learner interactions, data curation, latent space exploration, and conversion to action. However, the second factor is a major unaddressed challenge that has resulted in a situatedness gap in the literature.

The Situatedness Gap

There is a major gap in the current state of ML and creative activities when it comes to grounding these tools in the context of their applications. On one hand, a lion-share of efforts in machine learning communities is focused on developing novel algorithms and improving their efficiency (Simard et al., 2017). On the other hand, there are fewer resources dedicated to informing the design process and the end results with inputs from the end-users and the contexts in which users will operate these tools. Studying the impact of these tools on design, practice, and the labor is another lesser-studied topic in this domain.

Situatedness gap is a well-documented phenomenon. Diana Forsythe—an anthropologist and a Science and Technology Studies (STS) scholar—provides us with first-hand observation of artificial intelligence and "Expert System" researchers in action in the late 20th century in her book "Studying those who study us" (2001). She explains the AI researchers' efforts for leveraging "knowledge acquisition" methods— i.e., surveys, interviews, and observations—to "extract knowledge" from the domain experts to build "knowledge bases". These "knowledge bases" were the raw material for the AI experts to build "expert systems". Throughout this process, there is a strong tendency to detach the expert users from the tool development process. A consequence of this approach is the ignorance of expert users' personal inputs, social ties among the experts, as well as information about the physical context of their practice.[24]

In a typical AI problem solving process, it is up to the AI specialists to decide on keeping details that they register as "relevant" in favor of making a simpler model (Brooks, 1991, p. 142). Such an abstract representation of the context tends to factor out the dynamic coupling between the system and its world and eliminates various aspects of perception and motor skills of an expert user. Thus, it is not as useful as continually referring to the real world, the immediate context (Brooks, 1997 as cited in Dreyfus 2007), and the expert. When creative users repurpose these tools in their work, these issues bleed into their creative process and intensify the situatedness gap.

While using the libraries and toolkits which were introduced earlier renders machine learning toolmaking more accessible for user with limited knowledge of ML, it is still a major challenge to address the contextual aspects of design. Design practices are situated in their physical contexts and thrive in their community of practice. Most of the efforts on the intersection of design and machine learning are ignoring this point and stripping design tools from the contextual relationships between 1) toolmakers, 2) tool, and 3) design problem that they are addressing.

Training a learner on data sets which are scrapped from online databases or processed from building information modeling (BIM) data sets neglects the situated characteristics of the design tasks. These characteristics can be physical—such as material behavior, tool affordances, and tool limitations—and social— i.e., toolmakers' perception of the tool, subjective assessments of its performance, or interactions between toolmakers in the process of toolmaking.

A situated approach to toolmaking aims to distance from the abstract and decontextualized approach and help the expert users to take command of the process. It allows the users to make tools that are grounded in

---

[24] I will elaborate on the social and physical ties in the next chapters. To depict a brief image, the social ties may include the interactions between the expert user and its collaborators as well as colleagues. The physical ties may include material behaviour, tool limitations and affordances, in addition to physical environment factors such as dimension.

real-life scenarios and addressing challenges that an expert user faces on its routine workflows. In this process, the context, the task, and the expert users are governing the inclusion or exclusion of data. The situated approach aims to set the expert user as the source of data and advocates data collection methods that can be used in the real context of a practice.[25]

For example, these system can be tuned and set to collect hand gestures of a painter while it is in its studio painting with brush, paint, and canvas, or to record the camera motions of a cinematographer on the set, or to observe dancers' motions in a performance or rehearsal. In each case, the user can improvise, repeat, modify, or remove actions to sculpt the data set, one sample at a time. That is a paradigm shift from the conventional approach in which an ML expert would decide on this matter. I expect that this approach reduces the chance of critical data being ignored in favor of a simplicity and abstraction.

In this approach, data is treated as a malleable "design material" in the hands of expert users. This new form of design material helps users interfacing the ML backend by generating, collecting, and curating training data to incrementally shape the learner's behavior. As the expert users play more with their malleable design material, the resulted data set becomes more skewed and biased towards their own special and unique design tastes. The resulted ML tool will be loaded with personal judgments, preferences, and subjective measures that make it unique to the person who made it.

This process also leads to a gradual evolution of the conception of the tool in expert users' minds. This conception gradually forms during the interactions with the learner in a context which is very familiar for the expert user.[26]

Situated Machine Learning Examples

There are a few attempts to incorporate the physical characteristics of materials in the process of digital fabrication. Researchers are developing methods to incorporate manual dexterity, material behavior, and tool characteristics into their ML tools to improve robotic fabrication methods for wood carving and metal forming (Brugnaro & Hanna, 2017; Rossi & Nicholas, 2018).

Giulio Brugnaro, a Ph.D. researcher at the University College of London, developed a learning system to inform the robotic fabrication workflow on the specific properties and behavior of the material. In this system, an expert user demonstrates several samples of carving a wooden piece with a chisel. The demonstration is recorded as a sequence of motions in space with six degree of freedom—representing both location and orientation of the tool at each time stamp. These examples were later augmented by a series of machine generated samples to form a learning data set. Brugnaro designed a neural network to map these chisel motions into the carves that they made on a piece of wood. The final ML models could predict the result of a chisel motion on a specific piece of wood, and inversely, predict the necessary chisel motion to create a given carve on the wooden piece.[27]

In his workflow, the robotic arm, fabrication system, data collection tools, and material feedback sensors are intertwined to inform the learner about the physical ties between the expert user, the tool, and the

---

[25] Some of these methods may entail design and fabrication of bespoken tools to collect data.

[26] It is worth mentioning that even this approach is not completely free of the biased and flaws that Forsyth discussed.

[27] They describe this project as " … a robotic fabrication system where the instrumental and material knowledge of skilled human craftsman is captured, transferred, robotically augmented and finally integrated into an interface that make this knowledge available to the designer" (Brugnaro, Figliola, and Dubor 2019, 151). An underlying assumption of Brugnaro's framework is that an expert's knowledge can be captured, stored, and transferred, which is a source of debate and I have previously discussed this matter in one of the qualifier's essays.

material. He describes the system as a "… soft systems, both adaptable and continuously evolving, whose dynamism is constantly fed by a flow of information" (Brugnaro, Figliola, & Dubor, 2019, p. 135).

Figure 2- Data collection apparatus, training samples, and execution of learned motions, images from (Brugnaro & Hanna, 2017).

This project is mostly focused on the material-tool behavior and leaves the personal or social context unaddressed. For instance, the evolution of toolmakers' conception of the tool during the toolmaking process is not a primary focus. The toolmaker's subjective evaluations are also missing from this project, once the learning samples are generated, the toolmaker loses its control and agency over the augmented samples, learning direction, or evaluation of the results.

Rebecca Fiebrink, a computer scientist, pioneer of Art and ML, and reader at Creative Computing Institutes at the University of the Arts London, on the other hand has focused on the agency of the toolmaker, not only during the sample generation phase, but also during the training process. She developed a meta-instrument that let an artist develop a musical instrument in collaboration with an ML expert through interactive supervised machine learning (Fiebrink, 2011). Wekinator—an open-source software for real-time interactive machine learning developed by Fiebrink in 2009—[28] shapes the ML backend and let the artist design a workflow in which the ML expert could moderate the technicality of the learner while the artist was responsible for creating samples, evaluating, and assessing the tool's performance (Figure 3).

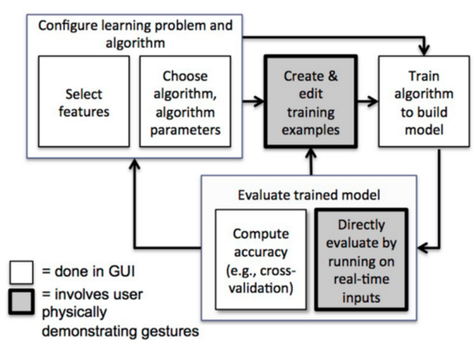

Figure 3- Interactive learning workflow in Wekinator, from (Fiebrink, 2017, p. 163)

As a case study, Fiebrink collaborated with a professional composer/cellist to make a classifier that was able to recognize standard bowing gestures for live performance or composition. Throughout this collaboration, the expert user contributed in two major ways: 1) Providing the learning samples by playing her cello using a custom-made bow. Fiebrink demonstrated a range of techniques on the real instrument and recorded them as a sequence of multi-dimensional data by the sensory systems installed on the bow, 2)

[28] http://www.wekinator.org

Introducing her perception of the tool, personal feelings, and subjective assessments as a part of evaluation and feedback process. This project is one of the few examples that manages to incorporate both physical and social features of the creative task in the process of toolmaking.[29]

Discussion

Projects such as Magenta, Runway, and Wekinator are helping toolmakers to focus on the creative process of toolmaking rather than technical challenges that working with ML models entail. They reduce the necessity of engagement with the technical details while focusing on exploring the process and the outputs; thus they (partially) successfully cast the ML toolmaking as a design problem rather than a merely technical one. However, each of these projects has focused on one of the challenges of machine learning toolmaking for creative activities. There is still a wide area to explore when it comes to integrating the toolmaking apparatus with the power of ML generative models with situated interactive learning and the sequential/multi-dimensional data.

To address this gap, this research proposes a situated approach to ML-based creative computing toolmaking. It focuses on the importance of creating systems capable of incorporating social and material aspects of creative practices, such as the links between the 1) human agents—i.e., toolmakers, users, and audiences, 2) the tool, and the 3) the practice on the one hand, and the process of designing, training, evaluating, and using such tools on the other. These contextual ties include elements such as human agents' perception of the tool, their subjective assessments, material behavior, and the limitations and affordances of the physical tools.

[29] In her 2017 paper, "Machine Learning as Meta-Instrument: Human-Machine Partnerships Shaping Expressive Instrumental Creation", Fiebrink elaborates the idea of using interactive machine learning as a meta-instrument. She discusses how supervised learning can be leveraged to design new tools in real-time for creative activities and emphasizes on the relationship between the builder and the toolmaking procedures as a key factor to understand the new instrument. She argues that this can result in "… an exploratory, playful, embodied, and expressive …" toolmaking process (Fiebrink, 2017, p. 137).

Research Elements

Question

The primary question of this research is:

- How can creative users utilize machine learning to address the gap in the current state of creative computing toolmaking that sets apart end-users and the context of the practice from the toolmaking process?

To address this question, I will also investigate the following questions:

- What are the properties of an accessible ML for creative users?
- Which ML techniques can address the situatedness gap in ML?
- What are the characteristics of a situated approach in ML-based creative computing toolmaking?

Hypothesis

I hypothesize that a framework that is built on 1) the situated approach to ML toolmaking and 2) accessible and user-friendly machine learning system can open new opportunities for developing situated creative computing tools. This framework should help creative users incorporate spatial, material, and social registers of their practice to the data collection, learning process, and eventually their creative computing tools.

Such an inclusive tool making framework can put the creative users at the center of toolmaking process and enhance the development of a constructive relationship between the creative user, its tool, and the context of a practice, and address the decontextualization problem in the toolmaking process.

Objectives

The proposed research aims to study the affordances of situated approach in ML-based creative computing toolmaking and develop a framework to incorporate spatial, social, and material connections between the toolmakers, their toolmaking apparatus, and their craft. The primary objective is to develop a framework allowing domain experts to build ML-based tools with respect to the context in which the tool is being created and used. The contextual registers vary from task to task, and can be physical—i.e., material properties, tool characteristics, external forces, and/or social —i.e., conception of the tool, subjective assessment and metrics, collaboration between users, personal preferences or approaches.[30 & 31]

Approach

Crafting ML models is an abstract procedure that requires a deep understanding of the mathematical background and machine learning algorithms. This research aims to proposes a method that make ML models more accessible and simplified for domain experts to build their tools without forcing them to

---

[30] To illustrate an example, in an ML-based painting tool, the framework can help the artist to integrate different brush, paint, and canvas properties (physical registers) as well as its personal preferences in color combination, brush gestures, and order of strokes (personal registers) and integrate its colleagues input in the form of training samples or learning steering hints.

[31] From a broader perspective, this framework does not need to be limited to creative practices. The expectation for this research is to contribute to the development of tools that serve various human-machine hybrid work scenarios where a skilled human agent cooperates with an AI equipped machine on a given task. For example, tasks that require hand dexterity such as delicate manufacturing and assembly tasks or even trivial routine fabrication tasks.

engage with the complexities of the backend ML systems. In other words, this framework should enable expert users to be both the toolmaker and the user of their tool. This framework is built upon the *expert user- learner interaction* model to create context-aware creative computing tools.

Compared with current practices in ML, the proposed framework should help toolmakers in two major capacities:

1) Incorporating personal and social aspects, including but not limited to:
   a) Enriching the training data set: Curating a hybrid of available data sets and personal samples which helps to introduce desired features, steering the learning direction based on personal preferences and subjective assessments;
   b) Enriching the evaluation loops: Providing personal and subjective evaluation and feedbacks by the expert user in an interactive process;
   c) Collaboration: Working in collaboration on the process of tool development through collaborative data production and evaluation.
2) Integrating material and physical aspects, including but not limited to:
   a) Demonstration instead of programming: Informing the fabrication process through demonstration, rather than hard coding;
   b) Physical integration: Integrating material behavior and tool affordances by learning from real samples rather than simulations;
   c) Data augmentation: Generating hybrid fabrication techniques by curating a hybrid of produced learning samples and physically augmented samples.

Scope

One of the challenges of using ML in creative practices is the broad scope of activities and data types. To set this research on the boundaries of a Ph.D. dissertation, the scope of the proposed framework is narrowed down in respect to 1) activities, 2) data type, and 3) application:

Activities: This framework will focus on the primary *actions* that can be combined by the expert users to accomplish a more complex *task*. These actions can range from rough fabrication activities—i.e., basic subtractive or additive manufacturing—to primary elements of a performance art that can be actuated using robotic arms. As illustrated in Figure 4, this research doesn't address the process of generating novel creative pieces, nor an autonomous process to plan sequence of actions to embody such pieces.[32] But, it is focused on providing an expert user who already has addressed these two steps with a tool to embody its artistic impression.

[32] This proposal intentionally distances itself from these two phases, as they more inclined towards autonomy and Computer Creativity.

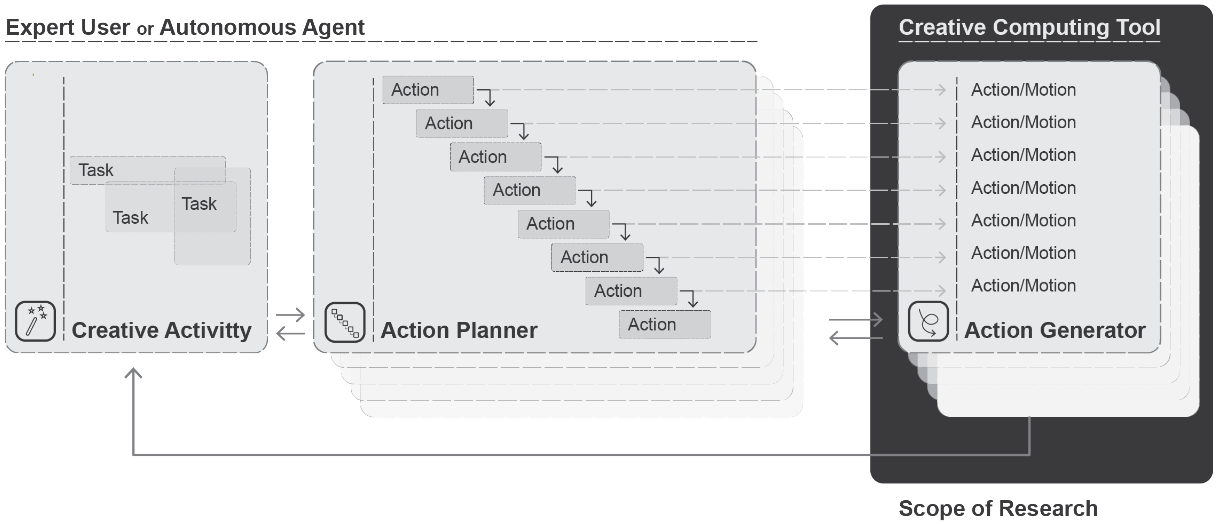

Figure 4- Scope of research with respect to activities

Datatype: Based on these domains, the data types are limitted to two categories that foster a wide range of applications in the field of design and performative art: 1) three-dimensional data —i.e., point clouds, mesh, and voxel models; and 2) sequential data —i.e., motion capture streams, music, and text.

Application: On the application side, it can serve as a generative tool to create new actions based on the previously seen samples.

Limitation: Considering the wide range of creative activities, it is virtually impossible to develop a framework to address them all. Accordingly, the framework's scope of functionality is limited to specific activities that it is adapted for.

Novelty

The body of literature on situated and accessible machine learning is sparse. There are a few efforts to partially address various topics of this proposal. This research aims to explore multiple topics that have not been simultaneously addressed in the literature of machine learning in creative computing:

3D and sequential data: As discussed earlier, this research will focus on three-dimensional and sequential data —i.e., motion capture streams. It is a departure from two-dimensional and pixel-based generative models that are currently prominent in the creative machine learning literature.

Data as Interface: Compared with classic machine learning techniques, the proposed framework will focus on data as a malleable element in the hands of toolmaker and the primary interface between the toolmaker and the learner.

Latent Space Exploration: This research also aims to let toolmakers explore the latent space of the generative models as a critical stage of toolmaking process. The latent space navigation let the toolmaker explore new outputs and affordances in the tool. This is another form of interaction with the learner through data, although this time it is indirect and abstract. The toolmaker can explore the latent space and generate unseen samples, combine, and interpolate between them to build a conception of the tool's abilities and design new feature with the tool.

Generative Models for Actions: There are multiple examples of ML tools that serve primarily as a discriminative tool—i.e., classifying gestures and associating each class to a specific output. This research aims to take advantage of generative machine learning models using many-to-many mapping.[33] In many-to-many mapping, a high-dimensional input vector—i.e., a complex hand gesture in space, a video stream, or a point cloud—will be collected as the input for the learner to generate a high-dimensional output vector of signals—i.e., a sequence of robotic motions, an image, or stream of video.

The generative model can be trained as an action-to-action (A-2-A) or action-to-result (A-2-R) model. In action-to-action, the model will be trained on a series of actions to generate new actions. This is useful when the action itself is the goal of the creative practice. For instance, in choreography the model can be trained on performers' motions and generate new ones.

On the other hand, in A-2-R—or equivalently R-2-A approach—the model will be trained on a series of actions and their results simultaneously. It will learn to predict a result from an action or vice versa. For example, the model can be trained on a series of brush motions as well as the strokes they leave on a canvas. Then it can generate a brush motion that will leave similar strokes on canvas or observe a stroke on canvas and generate a corresponding brush motion.

In some use-case scenarios a stand-alone action is the goal of activity. For example, in cinematography a single camera motion is sufficient for the cinematographer. In some other activities, a sequence of actions leads to the activity's goal. In such cases, an intermediator agent is required to convert the transition between the initial state and the final state into a series of discrete actions. For example, in painting an agent (the artist or an AI agent) should decide on the sequence of brush strokes to convert a blank canvas into a finished painting. This research is not focused on the intermediator agent nor an autonomous generative system to serve as an "AI artist". The primary objective of this research is to help an expert user to execute its own plan—either a predefined or an exploratory one—using the action space provided by the generative tool for actions. It is the expert user's due to utilize this tool to shape its creative expression.

Physical Platform: Practices that are inseparable from their physical representation will require a method to translate digital signals into physical actions. To fulfil this requirement, the use of industrial robotic arms is proposed herein. Industrial robotic arms are programmable and flexible mechanical machines that have gained attention of architects, designers, and artists in several research institutions across the globe during the past decade. These machines can precisely actuate motion instructions while carrying various tools and payloads. This flexibility and familiarity in the research community renders these machines as a capable candidate for this research.[34]

Deliverables

The primary deliverable of this research is the framework for a context-aware approach to creative computing toolmaking. The second deliverable is an implementation of the framework in the format of a software toolkit. This toolkit will be used to study and test the framework in action. Both framework and

[33] On the other hand, Fiebrink's bow-gesture classifier is an example of many-to-one mapping system. It collects a high-dimensional vector of input data from various sensors—accelerometers and pressure sensors—to classify them in a one-dimensional vector that defines the class of gesture.

[34] Ultimately, a collaborative robotic arm—robotic arms that are designed to work in close collaboration with human users and are equipped with various safety precautions such as UR robotic arms or ABB's YuMi—would be the better choice.

toolkit will be discussed in detail in the Methodology chapter. Moreover, throughout the user studies, one or more instances of toolkit use-cases scenarios will be produced.

Related works at CMU School of Architecture

During the past few years, several local artists and craftsmen have been invited to collaborate with students and ongoing research projects in dFab (Figure 5, left), most recently, a group of artists, including graffiti artist, wood printers, and plastering experts (Figure 5, right). I will leverage these connections to find and invite artists for collaboration.[35]

One point of departure between these previous efforts and this proposal is the situated approach and focus on generative models. In the previous studies, the craftsmen and artists were invited for (a) "recording session(s)", where they demonstrate their work in the "lab" environment. On the other hand, this research aims at machine learning toolmaking process, interactive learning process, and situatedness. This may require studying and performing the demonstration in the context that the craft is situated, i.e., the craftsman's workshop or artists' studio.

Figure 5- Robotic plastering (left) and graffiti (right) (Bard et al. 2014).

Methodology

To test the hypothesis, I propose a four-step approach: 1) defining the framework, 2) toolkit implementation, 3) user studies, and 4) analysis. I will first elaborate on the details of framework and toolkit, then I will introduce a series of steps and studies to implement and assess them in various use-case scenarios.

Framework

The first step of this research will be dedicated to defining the theoretical framework that specifies the main characteristics, features, and boundaries of the situated toolmaking approach for creative computing that entails a context-aware perspective to ML in design, art, and making.

The proposed framework should empower the expert users to 1) collect contextual registers (spatial, social, and physical), 2) train a generative model, and 3) navigate and explore the possible motion/making space. To support these three features, the framework should take these factors in consideration:

---

[35] For more information, please refer to (Bard et al., 2014) and the web site of Human Machine Virtuosity course in Spring 2019 (https://courses.ideate.cmu.edu/16-455/s2019/)

Accessibility: Domain experts are not necessarily machine learning experts. They may have limited prior exposure to programming, mathematics, and machine learning. The framework should render the process of toolmaking accessible for these expert users.

Supporting Bespoken Situated Data Collection: In creative practices, expert users need to generate bespoken training data sets to customize the learner to their own preferences. The framework should support the data generation procedure and provide methods for data pre-processing and storage. As we look for a situated approach, the data collection process is focused on contextual registers. These contextual registers are usually multi-modal and high-dimensional—i.e. complex sequence of hand gestures and effects of tool over material. The data collection and storage system should encompass data pipelines and pre-processing workflows.

Train, Feedback, Correction: The process of training is interactive and iterative. The expert user needs to interactively engage in an iterative cycle of "train, feedback, correct" actions. This is an open-ended effort where the expert user may want to refresh the cycle at any stage of the tool's life cycle.

Generation: The resulted tool should help expert users not only reproduce the original demonstrated actions, but also it should provide functionalities to explore new action spaces which are derived from the training data set. Thus, the tool should enable the expert user to navigate the learned space for generating new instances.

To address these four factors, I utilize a series of method listed in Table 1 and discussed further below.

Table 1- Features and methods to address each feature

	iML	*LfD*	*AE & VAE*
Accessibility	+	+	-
Bespoken Situated Data Collection	+	+	-
Train, Feedback, Correction (TFC)	+	+	-
Generation	-	-	+

Accessibility

There are various approaches to make ML more accessible to users with limited knowledge of ML. One approach is to decouple the knowledge of ML algorithms from the process of teaching machines. The key concept in this approach is to let the expert user interface the learning system not through code, but through data and observing the reactions that the learner generates in response to the user inputs.[36] It is expected that the combination of this approach with a user-friendly interface and readable visualizations will help expert users—who have a comprehensive understanding of the problems' semantics—to directly engage in the learning process and understand the behavior of the machine and its learning procedure.

Interactive Machine Learning (iML) is one of the approaches that contributes to the accessibility of an expert user-learner system. iML is particularly useful for creative applications and developing custom-made tools and shines the best when we have a creator who can generate reliable training examples (Fiebrink,

[36] This approach is closely related to Machine Teaching (MT). MT is a paradigm shift in ML; while ML is focused on creating algorithms and improving the accuracy of the learners, MT emphasizes on the importance of teachers and aims to improve the efficiency of the teachers given a fixed learner. In MT, the teacher debugs a model by 1) inspecting, 2) adding and editing "knowledge" such as labels and features, 3) train, and 4) testing. A teacher performs these actions "without being required to understand the runtime function space, learning algorithms, or optimization" (Simard et al. 2017, 4).

2019, p. 3). The creator who is a domain expert in this case, improves the model's performance by constantly providing new training examples until the desired performance is achieved. In iML, the computer is a part of the human design process, rather than the human being in the loop of an algorithmic process (Gillies et al., 2016).

In this process, the learner is either fixed or has a limited number of parameters for fine-tuning. An expert user controls the learning progress based on its subjective judgments through a series of rapid train-feedback-correct cycles. It chooses when and what to be provided as new training samples after observing the outputs in each cycle (Amershi et al., 2014).

Since each expert user has a unique understanding and interpretation of evaluation metrics and progress benchmarks in comparison with other expert users and ML experts, the outcomes of iML are tailored to the unique taste of the expert user who has trained it.

Bespoken Situated Data Collection

One of the bottlenecks of ML—and specifically deep learning— is its heavy reliance on abundant and accessible training data sets. However, for developing custom-made creative computing tools, such data sets might not be easily accessible or even exist.

While there are various methods to mitigate this problem, I suggest using learning from demonstration (LfD) method.[37] LfD is an optimal choice for training systems that require interactions with users in tasks with a high level of manual dexterity. In such cases, an expert user feeds the system with training data, observes the results, and provides feedback. LfD has been used for a wide range of learning scenarios from playing video games and puzzles (Packard, 2017), to playing football in Robot World Cup (Floyd, Esfandiari, & Lam, 2008), object manipulation, controlling robots (Argall et al., 2009), and subtractive manufacturing (Brugnaro & Hanna, 2017).

To shape a situated learning system, the expert user needs to command three procedures: 1) data curation in the context of practice, 2) learning process, and 3) evaluation. LfD helps expert users to moderate the first one, while iML contributes to the second and third procedures.

In the data curation phase, LfD helps users to produce learning samples in an environment which is either the real context of the task or closely represents such context. Thus, the contextual signals will automatically be embedded in the training data set without the risk of getting filtered by an ML expert. During the learning phase, in both iML and LfD, the user constantly guides the learner by subjective inputs and decisions based on its experience and knowledge of the task. These signals will be an integral part of the result, reflecting personal and social aspects of the expert user.

In both iML and LfD, a domain expert employs data as the interface, rather than a substrate[38] to provide samples, set precedents, and implicitly define the rules to form a design space in a learning system. The expert user interacts with the learner via data until a desired stylistic consistency is achieved.

[37] Other terminologies to address LfD includes learning from demonstration, learning by imitation, programming by demonstration, apprenticeship learning (Argall et al., 2009). It is considered as an intuitive and natural way of learning (Ontañón, Montaña, & Gonzalez, 2014), and aims to find a computational representation of the expert's actions given the observations of 1) expert's interactions and 2) the environment states (Packard, 2017).

[38] For an in depth discussion on data as interface please refer to (Cardoso Llach, 2017).

Generation

To generate new actions, I propose using a family of machine learning models categorized as Generative Models which are working based on the generative learning principal. Compared with other ML models, generative models do not passively observe the training data, but they build a perception of the data distribution (Deshpande & Purwar, 2019, p. 4). Generative Models can "explicitly or implicitly model[s] the distribution of inputs as well as outputs" (Bishop, 2006, p. 43). Once the distribution is formed, it can draw synthetic samples from the new distribution. These generated samples closely resemble the original training data.

AutoEncoders and Variational AutoEncoders

While there are various generative models, this research focuses on AutoEncoders (AE) and Variational Autoencoders (VAE) due to their stable learning process, efficient forward pass, and the level of control over the generation synthesis.[39]

Autoencoder is a generative model architecture that encompasses two neural networks that are chained together. The first network is called *encoder* or *recognition model*. It is being trained on the input data to learn the features and encode them in a latent representation space which is usually of lower dimension and referred to as the *bottleneck*. The *decoder* network, also known as *generative model*, is another network that receives the samples from the latent space and returns samples that closely represent distribution of the input data.

In the training process, the encoder tries to reduce the dimension of data and map it to the latent space z. The decoder aims to get the latent representation and reconstruct the original input data. In the process of encoding, some data will be lost as the input is squeezed into a lower dimension. When the decoder reconstructs the input from its latent representation, the outcome will not be identical to the input. The objective of an AE is to reduce the amount of lost data between the input and the reconstructed output. Generating a new sample is possible by feeding the decoder network with a random vector in the same format of the latent space.

AutoEncoder architecture fits the proposed approach due to its relatively simple architecture and stable learning process. It is an efficient architecture during both training and inference phase. This efficiency is a key factor in developing an interactive system with acceptable gap between user's input and returning results. However, a vanilla autoencoder is not the perfect solution. While theoretically generative decoder network should produce a sample from any correctly formatted random vector, the results might not be valid when the input signal differs significantly from those of the training data set. This problem makes AEs less stable and reliable as a generative tool.

Variational Autoencoders (Kingma & Welling, 2013) are another architecture in generative models. While the VAE architecture is resembling Autoencoder architecture, they are not the same (Doersch, 2016). Just like AE, VAEs have two networks, an encoder, and a decoder. However, in a VAE the latent space itself is a distribution, usually normal distribution. To generate a sample of the latent space, one can draw samples from this distribution. Thus, the process of drawing samples is more consistent and reliable. Thus, I propose to use VAE as the initial core architecture for the machine learning backend.

A generative model without proper interface and modes of interaction behaves as a black box. Users cannot comprehend the behavior of the model or predict the probable outcomes based on their inputs to the

[39] For a comprehensive discussion on the advantages of AE and VAE over other models such as GANs and Auto Regressive models, please refer to my qualifier essay on Generative Models.

generative model. Thus, for the generation phase, I propose to let users interactively manipulate the latent space of a generative model. This manipulation can be direct (Bidgoli & Veloso, 2018; Guan, 2018) or indirect through visual inputs (Park, Liu, Wang, & Zhu, 2019; J. Y. Zhu, Krähenbühl, Shechtman, & Efros, 2016) or using other interaction modalities, i.e., text (Reed et al., 2016; Zhang et al., 2017).

The proposed framework forms an end-to-end pipeline that relies on iML and LfD to interactively train a generative model in collaboration with the expert user. The expert user will interface this trained model using explicit or inexplicit signals to create new samples that can help him/her to accomplish a task. In the next section, I will discuss a methodology for implementing the proposed framework.

Challenges of small data set:

A potential challenge in the data collection process is the relatively small size of training data set that an expert user can produce in a given time. While state-of-the-art machine learning models are trained on "big" data sets, sourcing all samples from a single user can result in a significant bottleneck that may result in overfitting on the model. [40] To adders this issue, I aim to leverage two approaches: 1) increasing the dataset size (i.e., data augmentation), 2) leveraging learnt features (i.e., one-shot learning and transfer learning).

One probable solution to address this challenge is to utilize augmentation methods, where the amount of training data is increased using only the data that is already existing in the data set (Perez & Wang, 2017). For example, in an image data set, augmentation constitutes of a series of transformation procedures, such as rotate, pan, and scale to convert existing samples into slightly different ones that can be used to train the model.

The other approach to address the small data set challenge is to leverage what a model can learn in one situation to improve its performance in a new situation. In this case, instead of training the model from scratch upon any user's modification, we can utilize the trained model and what it has learned as a base point to enhance the next iteration of interactive learning. In one-shot learning instead of starting to learn from scratch, we utilize the information that has been learned from the previous samples—regardless their differences—to process new samples (Fei-Fei, Fergus, & Perona, 2006). It is also possible to transfer what has been learned in one situation to another situation and enhance the generalization in the target situation (Goodfellow, Bengio, & Courville, 2016).[41]

Toolkit

After the framework's boundaries and characteristics are defined, I will use it as the basis for developing a toolkit that can be modified to serve in various scenarios. The toolkit is expected to form an end-to-end workflow constituting of the data pipelines, ML algorithms, and API integrations with a user interface and set of user interactions.

---

[40] In comparison with popular toy data sets for machine learning practice, i.e., MNIST which has more 70000 samples, an expert user may only provide a few hundred samples per session.

[41] A possible challenge in this case is the fact that model may eventually "forget" what it had "learned" during the early cycles of learning or samples that it hasn't seen for a long time. For further discussion on this topic please refer to (Kirkpatrick et al., 2017).

The toolkit can be developed in two formats, 1) as a stand-alone service with APIs for various platforms, 2) as a plug-in for an already available platform. Decision on the format and of implementations will be made as the toolkit development proceeds.[42]

The toolkit will support three main phases of the toolmaking process: 1) collecting and processing data, 2) training machine learning back-end models interactively, and 3) generating actions. For each phase, various features will be organized as a series of modules to work in tandem and provide necessary features as illustrated in Figure 6.[43]

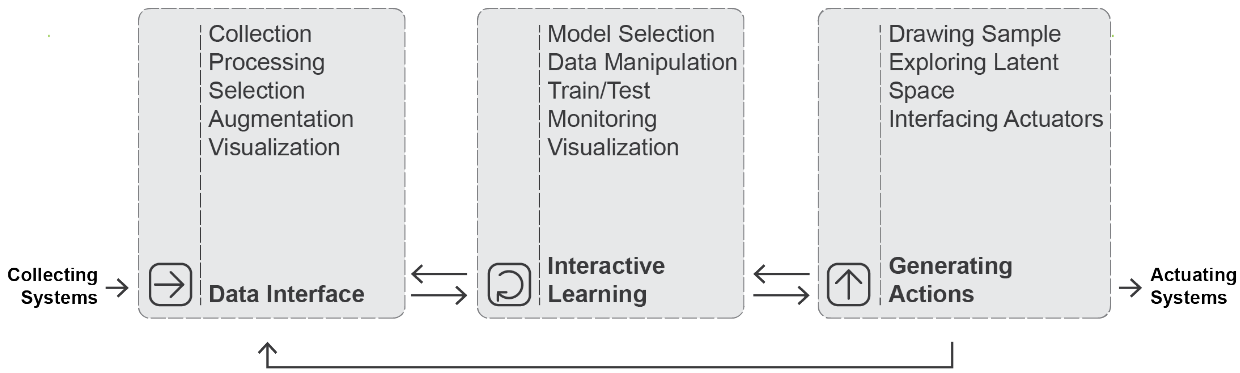

Figure 6- Prototype tentative modules

Data Interface:

The data interface module allows the expert user to interface various data collecting tools. This module includes basic data pre-processing methods as well as visualizing tools that help the user to inspect and analyze collected data. To compensate for the small size of data set, data augmentation methods will also be available and can be used when applicable.

Interactive Learning:

This module is dedicated to supporting an interactive learning process. It may include a library of machine learning models to select from as well various tools to manipulate, monitor and visualize the learning process. The user can iteratively move back and forth between these tools and the data interface to add new learning samples to the training data set.

Generating Action:

The features in this module will help user to navigate the latent space of generative model and draw new actions. This module is also responsible for interfacing actuation systems, i.e., a robotic arm.

---

[42] During the prototyping phase in the summer of 2019, a mockup of the toolkit has been developed as a stand-alone web-app. But as will be discussed later, it might be migrated to an available platform such as Runway for the ease of development.

[43] An early example of this toolkit is discussed in appendices under appendix Part II: Toolkit.

User Studies

A series of user studies will be conducted to study the affordances and limitations of the framework and toolkit as a toolmaking apparatus in professional practice.

During these studies, two expert users from different domains of creative practices will be invited to use the toolkit to develop tools for their own practices.[44] The domains will be selected in a way to cover both action-to-action mapping and action-to-result mapping. The toolkit will be updated based on the feedbacks from each study and might be updated to match each field. Table 2 lists a series of possible professional practices that might be the subject of these studies:

Table 2- Potential professional practices for user studies

Field	*Practice*	*Type*
Building Trade	Surface finishing	A-2-A
	Carpentry	A-2-A
	Welding	
Design Activities	Model making	A-2-A
	Drawing	A-2-A
Art	Painting/Drawing	A-2-A
	Sculpting	A-2-A
	Choreography	A-2-M
	Cinematography	A-2-M

Hypothesis

The hypothesis in these studies is that the toolkit can help an expert user with limited knowledge of ML to:

1- Incorporate physical characteristics of its tool, material system, and physical context of the practice in the process of tool making;
2- Integrate some of the personal and subjective preferences to the process of toolmaking;
3- Include inputs from other collaborators and colleagues both in the toolmaking process as well as applying the tool in action;
4- Explore action space of the tool and generate new ones.

Task description

The expert users will use the toolkit to develop a tool for their fields of practice. The study will be a semester-long process, were expert users and I will meet over multiple sessions for on-boarding, data gathering, training, and debriefing. The length and frequency of sessions will be decided in a case-by-case fashion.

These sessions will cover these activities:

1- **On-boarding**: As a pre-experience step, the expert user will be introduced to the framework and the toolkit. At this stage, the intention, objectives, and deliverables will be shared with the expert user.
2- **User studies**: In this step, the goal is to gain a better understanding of the experts' field of practice. To have a better grasp on the topic, this phase combines traditional data collection methods, i.e. interview and surveys in combination with in-depth observation and interaction with the expert in

[44] As these studies require data collection from live participants, proper documentation to acquire Internal Board Review (IRB) will be prepared and submitted in advance.

the practice.[45] The outcomes of this step will contribute to fine-tune and revise the data pipelines, learning algorithms, and models of interaction.

3- **Briefing and training**: The expert users will be trained to use the updated model. Throughout the process, additional training and assistance will be available upon the participants' request.
4- **Data collection and toolmaking**: The expert user will use the toolkit to build the tool. This process will span across several sessions and multiple iterations. Changes to the toolkit will be applied based on the expert user's request. These two procedures will run simultaneously, and accordingly inform each other.
5- **Performance**: The expert user will use the tool to perform a series of tasks in their domains.

Performance

The performance can be executed in various shapes and scenarios. The final decision overt the detail of each performance will be determined in collaboration with the expert users and may be one of the scenarios described below:

1- Pre-defined goal:
- In this scenario, the expert user has already developed a concept for a piece as well as a plan of actions to execute it.
- The artist utilizes the trained model to generate sequence of actions that will eventually embody the concept.
- Example of single action scenario:
  - A cinematographer who already has a story board and a set of characteristics of the scene, uses a model that it has trained to generate new camera motion.
- Example of complex action scenario:
  - A painter has a sketch of the painting and a sequence of brush strokes to paint it. It uses the trained model to generate motions for a robot-mounted brush to paint it.

2- Improvising with the tool:
- This scenario focuses on the process of exploring the affordances of the tool and reflecting on that to create novel art pieces.
- The artist repeatedly uses the tool to generate new actions and play with the latent space to fine-tune the actions until a desired effect is achieved. The artist combines these actions to develop a swatch of actions and utilize them to "improvise" a new piece of art.
- For example, a carpenter who trained a tool for wood carving will play with the latent space to generate new actions that can match a new type of wood with different characteristics. Then uses these actions to gradually carve a desired pattern on a piece of wood.

Objectives

There are three primary objectives for this study:

1- To study and test the affordances of the toolkit in a professional practice scenario.
2- To study the process in which an expert user comprehends the framework and use the toolkit to form a productive interaction with learner in the process of toolmaking.
3- To study the affordances of the tool to represent physical and social registers in the outcome.

[45] For this step, I will personally join the expert users to work on separate projects to achieve a better understanding of their workflow, behavior, and logistics. Such understanding is not attainable using conventional language-based methods such as survey and interview.

Data collection and observation

The study will primarily focus on observing the expert user's behavior in the process of 1) getting introduced to the toolkit, 2) interactively introducing data to the toolkit to define their design space, 3) using the toolkit to design a series of actions, and 4) accomplishing a task using a sequence of actions.

Users' behavior will be documented through qualitative and quantitative methods. After this phase, users' qualitative feedbacks will be collected through surveys and unstructured interview to study its experience with the framework and the toolkit:

Qualitative

The expert user will be subject to observation during the training sessions. The goal is to observe its experience with the framework and the toolkit. The subjects of interest are the learning process, source of questions, and parts that require further explanations.

During the task operation and after the completion, users will be subject to unstructured interview and survey. The goal is to observe the users' experiences in using the toolkit to make the tool and also using the tool in practice. The subjects of interest are toolkit usability, data input/output limitations, requested features, users' satisfaction.

Quantitative

Alongside the quantitative data collection, a range of quantitative measures will be recorded. These measures cover two main categories 1) training data, 2) documentation of the process.

Training data

The training samples will be recorded to train the model. The format and type of data will be crafted for each use-case scenario and may vary between the two observations. However, as I discussed in the scope of research it will be sequential and 6-DoF.

Data can be collected using various methods such as RGB or RGB-D cameras, motion capture systems, or sensor-based motion trackers.[46] The decision on the data collection method will be made case-by-case based on the level of intervention,[47] required accuracy, and the context.[48]

Documentation of the process:

During the data collection phase, variables such as number of samples, time spent on each sample, failed and success samples and etc. will be measured.

---

[46] RGB cameras are the most common type of cameras in the market. They record a two-dimensional representation of view as a grid of pixels. A combination of images from two RGB cameras can be used to reconstruct a 3d representation of space. RGB-D cameras collect the RGB data and use another sensor to capture a depth-map scan of the space, these two channels of data can be combined to form a colored depth-map. Motion capture systems use an array of cameras and combine their images to accurately locate a series of specific objects, usually reflective markers, in the space. Gyros and accelerometers are sensors that can measure the motion of the objects that they are attached to. Each of these methods has its own advantages and limitations.

[47] For example, how much modification will be required to integrate the sensory system and expert user's tool.

[48] For example, collecting data for camera motion is easier using gyro and accelerometer sensors installed on the camera since many cameras are already equipped with these sensors—there is no need for adding extra sensors to the artists'—and the required accuracy is not high. However, brush strokes for a painting can be best captured using a motion capture system since the reflective markers are easier and lighter to implement while providing significantly more accurate samples.

Over the training phase, I will collect information on the training parameters, time spend on data selection, training, and evaluation, number of iterations, value for each hyper parameter, changes in the model in each iteration, and model performance.

Finally, upon the test process, I will measure the time spent on each designated task, number of fail and success cases, differences between the objective and performed task will be recorded.

These variables will be recorded either automatically as a part of the toolkit functionalities, or manually collected by observers. For a subset of these variables, users will be provided with questionnaires to report their perceived values at the end of each session in each phase. The collected data will be used to compare users' perception with the actual records.

Participants

The participant(s) will be invited from Pittsburgh's local worker, artist, and craftsmen organizations.[49] The individuals' names, fields of practice, and the number of participants will be finalized as the research progresses.

Prior related works

As of now, I am working on two studies that are aligned with this research proposal. While still in-progress, these projects are addressing two of the practices mentioned in Table 2:

1. Cinematography:
    - This is an ongoing project that I am developing in collaboration with Autodesk research team at Pier9 and introduced in Appendix B: Toolkit Prototype section;
    - The project is focused on developing a camera control tool for cinematography. The toolmaker, which is a cinematographer in this case, can collect a data set of cinematography examples to define the desired design space with its own samples (data curation). Then it can iteratively feed the learning model with a carefully selected subset of the training data to steer the learning process towards a specific style or genre (controlling learning process), while closely watching the outcomes by generating samples of camera motions (navigating the learned design space) until a desired consistency in that style is achieved.

Figure 7- Data collection process, Shoulder-mounted camera rig (left), generating samples (right)

2. Painting:

[49] For the prototypes, CMU students from different departments will be invited to contribute.

- This project is aligned with the ongoing projects that I am contributing to in collaboration with Dr. Eunsu Kang, Dr. Barnabás Póczos and NREC research team, Manuel Rodriguez and Andrew Plesniak.
- The study is focused on developing a painting tool. The artist can train the model interactively on various types of tools and mediums—i.e., various brush types, paints, canvas—as well as various techniques. During the data curation process, different sensors can collect various spatial and physical registers—i.e., hand gestures and outcomes. During the training process, the artist can interactively observe the learning direction and control it via new training samples until the desired style is obtained.

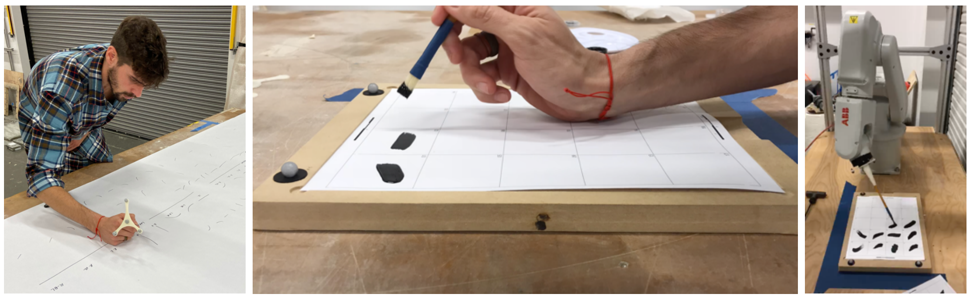

Figure 8- Early data collection for painting tool development and robotic playback

Logistics

For the data collection phase I will develop the required tools, mostly based on the equipment available in dFab, CodeLab, as well as IDeATe. For the training phase, desktop computers with state-of-the-art GPU card will be required. The(se) machine(s) will be maintained to the required software specs.[50]

[50] A live document with a list of machine learning models that can contribute to this study is available on this GitHub page: https://github.com/Ardibid/projects4creativeML

Timetable

Table 3- Time table (main tasks , subtasks , potential publication)

Tasks	F. 19	S. 20	Sum. 20	F. 20	S. 21
Framework development					
• Pre-design studies					
• System design					
Prototype/Toolkit					
• Design					
• Development					
○ algorithm alternatives					
○ data input pipelines					
○ data output pipelines					
○ System Integration					
• Tests					
Case Studies					
• Design					
• IRB authorization					
• User studies					
• Evaluation and analysis					
Writing					
Defense					

Extended Table of Contents

Part I: Thesis and Framework

Part I of this thesis will be an exploration to discover and explore the literature and flesh out the gaps, the research question, and the hypothesis. I will lay the foundation to introduce and define the proposed situated approach to machine learning in creative computing toolmaking and demonstrate its necessity and importance. This section opens the discussion to the broader landscape of Machine Learning as a means for creative work. I will discuss the background of Artificial Intelligence in the hands of creative users and proceed to review state-of-the-art ML in creative computing toolmaking from a critical standpoint. I will follow this topic by investigating the *Situated Gap* and its adverse effects.

The last section of this part will be focused on proposing a framework for context-aware machine learning in design, art, and making and check the validity of the hypothesis. This framework will constitute of these sections:

1. *Creative Computing Toolmaking*

In the first section in Part I, I will introduce creative computing toolmaking. I will discuss the main concepts, approaches, and perspectives to the topic in detail to clarify the topic, context, and scope of this research.

This section will also include a concise literature review of creative users' efforts to make computational tools to support their creative activities. I expect this literature review to serve as a prelude to the current state of Machine Learning in creative computing toolmaking.

2. *Machine Learning in Creative Computing Toolmaking*

This section will be dedicated to the current trends of using state-of-the-art machine learning research tools in creative activities.

3. *Situatedness Gap*

I will elaborate on the situatedness gap in this section. I will continuously go back and forth between this section and the previous ones to elaborate on the examples of the situatedness gap and emphasize its relevance and importance in the current state of ML in creative activities.

4. *Accessibility Gap*

Building on the studies from the previous section, I will focus on the obstacles that expert users confront with to interface machine learning tools. This section acts as a prelude for the next section, where I discuss potential solutions to bridge the accessibility gap.

I will introduce my hypothesis and elaborate on that to address these two gaps.

5. *Machine Learning Approach to Address the situatedness GAP*

Various machine learning techniques that can address the situatedness gap will be introduced in this section. I will use different examples from the literature to find the strengths and weaknesses of each method and make a compelling argument to form my framework.

6. *Framework Discussion*

The framework is the theoretical structure that explains my hypothesis to address the situatedness and accessibility gaps. This framework serves as a higher-level guideline to design and implement my methodology.

Part II: Toolkit

The second part of the thesis will explain the methodology to implement a toolkit based on the proposed framework. The primary objective of this section is to investigate the validity of the claims about the framework and its application in practice.

7. Toolkit Specifications

7.1.1. Human-centered approach

This subsection will introduce human-centered machine learning and discuss various methods to achieve it.

7.1.2. Interactive Machine Learning

I will introduce interactive machine learning and its background in this section, then I will proceed to discuss its application to address my research goal.

7.1.3. Learning from Demonstration

Learning from demonstration and its advantages to address the situatedness gap will be the subject of this section. I will discuss how LfD can integrate contextual registers to the process of toolmaking.

7.2. Data Pipeline

This section will be dedicated to the different data collection, data processing, and data management in the learning process.

7.3. Machine Learning Model's Architecture

Base on the previous discussions, I will propose an ML model and its associated user interface and user experience.

8. Methodology

8.1. User Studies

This section will be dedicated to the user studies on the framework/toolkit.

Part III: Reflection

The third part of thesis will be a reflection on the topics that have been discussed in the previous two sections. I will discuss the research process, the outcomes, challenges, new raising questions, next steps, and future works.

Bibliography

Amershi, S., Cakmak, M., Knox, W. B., & Kulesza, T. (2014). Power to the People: The Role of Humans in Interactive Machine Learning. *AI Magazine*, *35*(4), 105–120. https://doi.org/10.1609/aimag.v35i4.2513

Argall, B. D., Chernova, S., Veloso, M., & Browning, B. (2009). A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, *57*(5), 469–483. https://doi.org/10.1016/j.robot.2008.10.024

As, I., Pal, S., & Basu, P. (2018). Artificial intelligence in architecture: Generating conceptual design via deep learning. *International Journal of Architectural Computing*, *16*(4), 306–327. https://doi.org/10.1177/1478077118800982

Bard, J., Gannon, M., Jacobson-Weaver, Z., Contreras, M., Jeffers, M., & Smith, B. (2014). Seeing Is Doing : Synthetic Tools for Robotically Augmented Fabrication in High- Skill Domains. *ACADIA 2014 Proceeding*, 409–416.

Bidgoli, A., Kang, E., & Llach, D. C. (2019). Machinic Surrogates : Human-Machine Relationships in Computational Creativity. *Proceedings of 25th International Symposium on Electronic Art ISEA2019*. Gwangju, South Korea.

Bidgoli, A., & Veloso, P. (2018). Deepcloud the application of a data-driven, generative model in design. *Recalibration on Imprecision and Infidelity - Proceedings of the 38th Annual Conference of the Association for Computer Aided Design in Architecture, ACADIA 2018*, 176–185. Mexico City: IngramSpark.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. New York: Springer-Verlag.

Brooks, R. a. (1991). Intelligence without representation. *Artificial Intelligence*, *47*(1–3), 139–159. https://doi.org/10.1016/0004-3702(91)90053-M

Brugnaro, G., Figliola, A., & Dubor, A. (2019). Negotiated materialization: Design approaches integrating wood heterogeneity through advanced robotic fabrication. In F. Bianconi & M. Filippucci (Eds.), *Digital Wood Design: Innovative Techniques of Representation in Architectural Design, Lecture Notes in Civil Engineering book series* (Vol. 24). https://doi.org/10.1007/978-3-030-03676-8_4

Brugnaro, G., & Hanna, S. (2017). Adaptive Robotic Training Methods for Subtractive Manufacturing. *Proceedings of the 37th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA)*, 164–169. Acadia Publishing Company.

Cardoso Llach, D. (2015). *Builders of the vision, software and imaagination of design*. New York: Routledge.

Cardoso Llach, D. (2017). Data as interface: the poetics of machine learning in design. In C. Engemann & A. Sudmann (Eds.), *Machine Learning – Medien, Infrastrukturen und Technologien der Künstlichen Intelligenz* (pp. 195–218). [transcript-verlag].

Cardoso Llach, D., Bidgoli, A., & Darbari, S. (2017). Assisted automation: Three learning experiences in architectural robotics. *International Journal of Architectural Computing*, *15*(1), 87–102.

Chaillou, S. (2019). AI & Architecture: An Experimental Perspective. Retrieved April 4, 2019, from https://towardsdatascience.com/ai-architecture-f9d78c6958e0

Chris Donahue, Ian Simon, & Sander Dieleman. (2018). Piano Genie: An Intelligent Musical Interface. Retrieved May 7, 2019, from Magenta website: https://magenta.tensorflow.org/pianogenie

Davis, D. (2013). Modelled on Software Engineering: Flexible Parametric Models in the Practice of Architecture. Retrieved from http://www.danieldavis.com/papers/danieldavis_thesis.pdf

Davis, D. (2016). Evaluating Buildings with Computation and Machine Learning. In K. Velikov, S. Ahlquist, & M. Del Campo (Eds.), *Posthuman Frontiers: Proceedings of the 36th ACADIA Conference* (pp. 116–123). Ann Arbor: ACADIA.

Deshpande, S., & Purwar, A. (2019). Computational Creativity via Assisted Variational Synthesis of Mechanisms using Deep Generative Models. *Journal of Mechanical Design*, *August*, 1–14. https://doi.org/10.1115/1.4044396

Doersch, C. (2016). Tutorial on variational autoencoders. *ArXiv Preprint ArXiv:1606.05908*.

Dreyfus, H. L. (2007). Why Heideggerian AI Failed and How Fixing it Would Require Making it More Heideggerian. *Philosophical Psychology*, *20*(2), 247–268.

Dudley, J. J., & Kristensson, P. O. (2018). A Review of User Interface Design for Interactive Machine Learning. *ACM Transactions on Interactive Intelligent Systems*, *8*(2), 8:1-8:37. https://doi.org/10.1145/3185517

Eisenstadt, V., Langenhan, C., & Althoff, K. (2019). Generation of Floor Plan Variations with Convolutional Neural Networks and Case-based Reasoning An approach for transformative adaptation of room configurations within a framework for support of early conceptual design phases. In J. Sousa, J. Xavier, & G. Castro Henriques (Eds.), *Architecture in the Age of the 4th Industrial Revolution - Proceedings of the 37th eCAADe and 23rd SIGraDi Conference* (Vol. 2, pp. 79–84). Porto, Portugal: University of Porto.

Fails, J. A., & Olsen, D. R. (2003). Interactive machine learning. *Proceedings of the 8th International Conference on Intelligent User Interfaces*, 39–45. ACM.

Fei-Fei, L., Fergus, R., & Perona, P. (2006). One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *28*(4), 594–611. https://doi.org/10.1109/TPAMI.2006.79

Fiebrink, R. (2011). *Real-time human interaction with supervised learning algorithms for music composition and performance*. Princeton University.

Fiebrink, R. (2017). Machine Learning as Meta-Instrument: Human-Machine Partnerships Shaping Expressive Instrumental Creation. In T. Bovermann, A. de Campo, H. Egermann, S.-I. Hardjowirogo, & S. Weinzierl (Eds.), *Musical Instruments in the 21st Century* (pp. 137–151). Singabpore: Springer.

Fiebrink, R. (2019). Machine Learning Education for Artists, Musicians, and Other Creative Practitioners. *ACM Transactions on Computing Education*. https://doi.org/10.1145/3294008

Floyd, M. W., Esfandiari, B., & Lam, K. (2008). A Case-Based Reasoning Approach to Imitating RoboCup Players. *FLAIRS Conference*, 251–256.

Fonseca, R. W. Da, Didoné, E. L., & Pereira, F. O. R. (2013). Using artificial neural networks to predict the impact of daylighting on building final electric energy requirements. *Energy and Buildings*, *61*, 31–38. https://doi.org/10.1016/j.enbuild.2013.02.009

Forsythe, D. E. (2001). *Studying Those who Study Us: An Anthropologist in the World of Artificial Intelligence*. Stanford: Stanford University Press.

Gillies, M., Fiebrink, R., Tanaka, A., Garcia, J., Bevilacqua, F., Heloir, A., … Lee, B. (2016). Human-Centred Machine Learning. *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, 3558–3565. ACM.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. Cambridge, MA: Mit Press.

Guan, S. (2018). Transparent latent GAN. Retrieved March 31, 2019, from https://github.com/SummitKwan/transparent_latent_gan

Hanna, S. (2007). Automated Representation of Style by Feature Space Archetypes: Distinguishing Spatial Styles from Generative Rules. *International Journal of Architectural Computing*, *5*(1), 1–23. https://doi.org/10.1260/147807707780913001

Holzinger, A. (2016). Interactive machine learning for health informatics: when do we need the human-in-the-loop? *Brain Informatics*, *3*(2), 119–131. https://doi.org/10.1007/s40708-016-0042-6

Jabi, W., Chatzivasileiadi, A., Wardhana, N. M., Lannon, S., & Aish, R. (2019). The synergy of non-manifold topology and reinforcement learning for fire egress. In J. Sousa, J. Xavier, & G. Castro Henriques (Eds.), *Architecture in the Age of the 4th Industrial Revolution - Proceedings of the 37th eCAADe and 23rd SIGraDi Conference* (Vol. 2, pp. 85–94). Porto, Portugal: University of Porto.

Kingma, D. P., & Welling, M. (2013). Auto-Encoding Variational Bayes. *ArXiv Preprint ArXiv:1312.6114*, 1–14. https://doi.org/10.1051/0004-6361/201527329

Kinugawa, H., & Takizawa, A. (2019). Deep Learning Model for Predicting Preference of Space by Estimating the Depth Information of Space using Omnidirectional Images. In J. Sousa, J. Xavier, & G. Castro Henriques (Eds.), *Architecture in the Age of the 4th Industrial Revolution - Proceedings of the 37th eCAADe and 23rd SIGraDi Conference* (Vol. 2, pp. 61–68). Porto, Portugal: University of Porto.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., … Hadsell, R. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences of the United States of America*, *114*(13), 3521–3526. https://doi.org/10.1073/pnas.1611835114

Kyle McDonald. (2018). Dance x Machine Learning: First Steps – Kyle McDonald – Medium. Retrieved May 7, 2019, from Medium website: https://medium.com/@kcimc/discrete-figures-7d9e9c275c47

Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge university press.

Loyola, M. (2018). Big data in building design: A review. *Journal of Information Technology in Construction*, *23*(October 2017), 259–284.

Magenta. (2019). Magenta. Retrieved April 26, 2019, from https://magenta.tensorflow.org/

Mario Klingemann. (2017). Quasimondo | Mario Klingemann, Artist working with Code, AI and Data. Retrieved May 7, 2019, from http://quasimondo.com/

Mathias, M., Martinovic, A., Weissenberg, J., Haegler, S., & Van Gool, L. (2012). Automatic Architectural Style Recognition. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XXXVIII-5/*(March), 171–176. https://doi.org/10.5194/isprsarchives-xxxviii-5-w16-171-2011

Memo Akten. (2017). Learning to see: Hello, World! – Memo Akten. Retrieved May 7, 2019, from http://www.memo.tv/portfolio/learning-to-see-hello-world/

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill.

Monica Dinculescu, Jesse Engel, & Adam Roberts. (2019). MidiMe: Personalizing MusicVAE. Retrieved September 16, 2019, from Magenta website: https://magenta.tensorflow.org/midi-me

Newton, D. (2019). Deep Generative Learning for the Generation and Analysis of Architectural Plans with Small Datasets. In J. Sousa, J. Xavier, & G. Castro Henriques (Eds.), *Architecture in the Age of the 4th Industrial Revolution - Proceedings of the 37th eCAADe and 23rd SIGraDi Conference* (Vol. 2, pp. 21–28). Porto, Portugal: University of Porto.

Ontañón, S., Montaña, J. L., & Gonzalez, A. J. (2014). A Dynamic-Bayesian Network framework for modeling and evaluating learning from observation. *Expert Systems with Applications*, *41*(11), 5212–5226. https://doi.org/10.1016/j.eswa.2014.02.049

Packard, B. (2017). Policies for Active Learning from Demonstration. *The AAAI 2017 Spring Symposium on Learning from Observation of Humans Technical Report SS-17-06*, 513–519.

Park, T., Liu, M.-Y., Wang, T.-C., & Zhu, J.-Y. (2019). *Semantic Image Synthesis with Spatially-Adaptive Normalization*. Retrieved from https://arxiv.org/pdf/1903.07291.pdf

Peng, W., Zhang, F., & Nagakura, T. (2017). Machines' Perception of Space. *Proceedings of the 37th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA)*, 474–481. Retrieved from http://papers.cumincad.org/data/works/att/acadia17_474.pdf

Perez, L., & Wang, J. (2017). *The Effectiveness of Data Augmentation in Image Classification using Deep Learning*. Retrieved from http://arxiv.org/abs/1712.04621

Phelan, N., Davis, D., & Anderson, C. (2017). Evaluating Architectural Layouts with Neural Networks. *Proceedings of 8th Annual Symposium on Simulation for Architecture and Urban Design*.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2018). Language Models are Unsupervised Multitask Learners. Retrieved April 9, 2019, from OpenAI Blog website: https://www.techbooky.com/wp-content/uploads/2019/02/Better-Language-Models-and-Their-Implications.pdf

Rahmatizadeh, R., Abolghasemi, P., Behal, A., & Bölöni, L. (2016). Learning real manipulation tasks from virtual demonstrations using LSTM. *ArXiv Preprint ArXiv:1603.03833*.

Rahwan, I., Cebrian, M., Obradovich, N., Bongard, J., Bonnefon, J. F., Breazeal, C., … Wellman, M. (2019). Machine behaviour. *Nature*, *568*(7753), 477–486. https://doi.org/10.1038/s41586-019-1138-y

Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., & Lee, H. (2016). *Generative Adversarial Text to Image Synthesis*. https://doi.org/10.1017/CBO9781107415324.004

Roberts, A., Hawthorne, C., & Simon, I. (2018). Magenta. js: A javascript api for augmenting creativity with deep learning. *Proceedings of the 35th International Conference on Machine Learning*. Stockholm, Sweden.

Rossi, G., & Nicholas, P. (2018). Modelling A Complex Fabrication System, New design tools for doubly curved metal surfaces fabricated using the English Wheel. *Computing for a Better Tomorrow: Proceedings of the 36th International Conference on Education and Research in Computer Aided Architectural Design in Europe*, *1*, 811–820.

Rossi, G., & Nicholas, P. (2019). Haptic Learning: Towards Neural-Network-based adaptive Cobot Path-Planning for. In J. Sousa, J. Xavier, & G. Castro Henriques (Eds.), *Architecture in the Age of the 4th Industrial Revolution - Proceedings of the 37th eCAADe and 23rd SIGraDi Conference* (Vol. 2, pp. 201–210). Porto, Portugal: University of Porto.

Runway. (2019). Runway: Artificial Intelligence for Augmented Creativity. Retrieved April 26, 2019, from https://runwayapp.ai/

Sharp, O. (2016). Sunspring | A Sci-Fi Short Film Starring Thomas Middleditch - YouTube. Retrieved from https://youtu.be/LY7x2Ihqjmc

Simard, P. Y., Amershi, S., Chickering, D. M., Pelton, A. E., Ghorashi, S., Meek, C., … Wernsing, J. (2017). *Machine Teaching: A New Paradigm for Building Machine Learning Systems*. Retrieved from http://arxiv.org/abs/1707.06742

Stasiuk, D., & Thomsen, M. R. (2014). Learning to be a Vault - Implementing learning strategies for design exploration in inter-scalar systems. *Fusion, Proceedings of the 32nd International Conference on Education and Research in Computer Aided Architectural Design in Europe*, *1*, 381–390. Retrieved from http://cita.karch.dk/

Strobbe, T., wyffels, F., Verstraeten, R., Meyer, R. De, & Campenhout, J. Van. (2016). Automatic architectural style detection using one-class support vector machines and graph kernels. *Automation in Construction*, *69*, 1–10. https://doi.org/10.1016/j.autcon.2016.05.024

Tamke, M., Nicholas, P., & Zwierzycki, M. (2018). Machine learning for architectural design: Practices and infrastructure. *International Journal of Architectural Computing*, *16*(2), 123–143. https://doi.org/10.1177/1478077118778580

Tamke, M., Zwierzycki, M., Deleuran, A. H., Baranovskaya, Y. S., Tinning, I. F., & Thomsen, M. R. (2017). Lace Wall Extending Design Intuition through Machine Learning. In A. MENGES, B. SHEIL, R. GLYNN, & M. SKAVARA (Eds.), *Fabricate 2017*. UCL Press.

Wei, L., Ta, L., Li, L., Han, Y., Feng, Y., Wang, X., & Xu, Z. (2019). RAF : Robot Aware Fabrication Hand-motion Augmented Robotic Fabrication Workflow and Case Study. In J. Sousa, J. Xavier, & G. Castro Henriques (Eds.), *Architecture in the Age of the 4th Industrial Revolution - Proceedings of the 37th eCAADe and 23rd SIGraDi Conference* (Vol. 2, pp. 241–250). Porto, Portugal: University of Porto.

Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., & Metaxas, D. (2017). StackGAN: Text to Photo-Realistic Image Synthesis with Stacked Generative Adversarial Networks. *Proceedings of the IEEE International Conference on Computer Vision*, *2017-Octob*, 5908–5916. https://doi.org/10.1109/ICCV.2017.629

Zhu, J., Liapis, A., Risi, S., Bidarra, R., & Youngblood, G. M. (2018). Explainable AI for Designers: A Human-Centered Perspective on Mixed-Initiative Co-Creation. *IEEE Conference on Computatonal Intelligence and Games, CIG*, *2018-Augus*, 1–8. https://doi.org/10.1109/CIG.2018.8490433

Zhu, J. Y., Krähenbühl, P., Shechtman, E., & Efros, A. A. (2016). Generative visual manipulation on the natural image manifold. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9909 LNCS*, 597–613. https://doi.org/10.1007/978-3-319-46454-1_36

Appendices

Appendix A: ML in Architecture

Architects have explored the applications of machine learning in various capacities. On the one hand, a branch of machine learning in architecture efforts is focused on analytical and simulation tools. In these applications, objective, quantifiable, and verifiable data are being used to inform the process of decision making. On the other hand, another branch of machine learning research in architecture is focusing on creative applications and synthesis—i.e., using ML models to improve parametric generative models, exploring patterns and classification, accelerating simulation, and enhancing fabrication processes (Tamke, Nicholas, & Zwierzycki, 2018).

Architects use machine learning methods to facilitate the design, simulation, and fabrication of material systems (Stasiuk & Thomsen, 2014; Tamke et al., 2017), to simulate robotics fabrication procedures (Brugnaro & Hanna, 2017; Rossi & Nicholas, 2018), to design, evaluate, and analyze floor plans (As et al., 2018; Chaillou, 2019; Phelan, Davis, & Anderson, 2017), to perform spatial (Peng, Zhang, & Nagakura, 2017), stylistic (Hanna, 2007; Mathias, Martinovic, Weissenberg, Haegler, & Van Gool, 2012; Strobbe, wyffels, Verstraeten, Meyer, & Campenhout, 2016), and energy analysis (Fonseca, Didoné, & Pereira, 2013).

While a majority of these inquiries are thriving in academic environments, large corporations are also supporting—and benefiting—this trend of research. For example, WeWork is (was) interested in developing ML analytical tools for building and layout evaluation (Davis, 2016; Phelan et al., 2017). Also, companies which are not directly engaged in architectural design has also shown their interest in this field. For example, Raytheon BBN Technologies has invested in generating tools for conceptual design (As et al., 2018).[51]

Several researchers and practitioners have demonstrated that the cost of design changes increases as a project proceeds from pre-design towards construction. Computational approaches, such as parametric modeling and BIM, have demonstrated their ability to lower the design change costs by extending the window for making key decisions until later in the project life-cycle, where a better understanding of decision consequences is available (Davis, 2013). Machine learning tools can benefit from this curve-shift by 1) reducing the costs, 2) extending the design decision-making window, and 3) providing more intuitive and reusable interface. Despite these potentials of machine learning tools for professional architectural firms, they are the least active players in this field and it is hard to trace independent machine learning research in architectural professional practice.

Recent scholarship from the field of design and machine learning has shed light on some of the obstacles that impede the adoption of ML in architecture: 1) a major challenge is the wide and complex scope of architectural design activities. Thus, each category of design, development, construction, and occupancy requires a specific ML approach. 2) Data scarcity and lack of large, high-quality, and standardized data sets is another challenge. Although the growing rate of Building Information Modeling (BIM) adoption[52] as well as the availability of online art databases, improved the situation on this side. 3) Computation bottleneck is also another challenge, as architectural data are usually three-dimensional and computationally intensive. It renders data processing and preparation for a resource-demanding workflow. 4) Lack of clear quantitative metrics and methods to evaluate the results of a generative model is another challenge. Although the outputs of generative models can ultimately be evaluated by a group of designers, considering the scale of outputs, it can be a cumbersome task even through crowdsourcing. 5) Integrating ML with

---

[51] It is worth considering that corporations may not publicly publish their research efforts or intents in this field, potentially due to trade secret consideration or security precautions.

[52] BIM facilitates the data collection and processing due to its well-structured data management system (Loyola, 2018, p. 261). However, it is worth mentioning that relying on BIM databases inevitably imposes a significant bias on the learning data sets. This approach exclude building with no BIM models, i.e., older buildings, projects by small/local firms, and projects that cannot be modeled using BIM packages due to their special design features. Thus, any machine learning model which is trained on these data sets will be biased toward BIM-friendly designs.

classic computational design methods requires extra efforts to design and develop new workflows and procedures (As et al., 2018; Tamke et al., 2018).[53]

Appendix B: Toolkit Prototype

During the summer of 2019, I developed an early prototype of the proposed toolkit while working as a research intern at Autodesk Robotics Lab, at Pier9, San Francisco. The prototype served as a proof of concept for an end-to-end creative computing toolmaking pipeline, consists of data collection tools, data preparation methods, learning models, as well as a graphical user interface.[54]

Being a proof-of-concept, the toolkit needed to be narrowed down sufficiently to 1) reduce the technical complexity and freeing more space to elaborate the conceptual framework, 2) keep the prototype inclusive enough to be generalized later. Accordingly, we decided to focus on:

- 6 degree of freedom (6 DoF) sequential data: This is an inclusive area that addresses a wide range of trajectories in 3d space, i.e., sequence of a craftsman's hand gestures in space.
- An action-to-action approach: This eliminated the need for designing a conditional model to map actions to results and consequently reduced the complexity of the ML architecture.
- Camera motion planning: The rationale behind this decision was three-folded. First, it is an essential element of a cinematographer's camera language which makes it a legit candidate for toolmaking process; Second, it has all the characteristics of a 6DoF sequence (location, orientation, time-based behavior); Third, it poses fewer technical challenge as it doesn't require a direct physical contact between the camera and the subject. Thus, the generated motions could be executed on a robot-mounted camera.

Prototype domain

In this prototype, I aimed to study the possibility of learning an expert user's camera language, by training a generative model based on a series of samples provided by the same user. This generative model could later help the user to explore and generate new examples based on its camera language. The prototype was designed to let the user create training samples in its routine work environment with tools resembling its real tools.

During the data collection phase, the expert user, a cinematographer in this case, provides samples of its work. These samples include 5-second shots of a fixed subject using a wide range of camera motions and camera handling techniques. The collected samples went through a pre-processing pipeline before being fed into the learner. I used a subset of this data set to train various designs of autoencoder to serve as the backend of a web-based interface. The users who provided the training samples, or their collaborators, could later use this interface to generate new motions by modifying the latent space of a selected motion or combining multiple motions.

To accommodate this scenario, the prototype supports four major activities as described below: [55]

---

[53] Another point of discussion is the relationship between representation and the meaning. For example, it is naïve to assume that a mere pixel-based representation of an art masterpiece can convey its true artistic values. In another words, the abstract data representation is not a perfect vehicle to convey art.

[54] In this section, I use pronoun "I" when referring to myself and "We" to refer to the research team at Autodesk Robotics Lab.

[55] While these phases are listed in a sequential order, they were not necessarily followed in this order. A user may repeatedly generate samples and modify the learner to achieve a desired result without getting to the generating phase. This process can be illustrated better through one of our efforts: we repeatedly generate sets of training samples and trained our model based on them. Then based on the results, we returned to the data processing phase and updated our

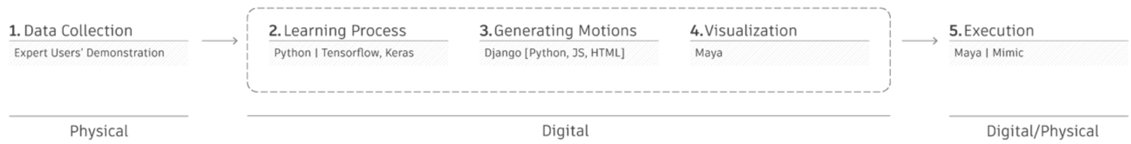

Figure 9-Toolkit development workflow

Data collection

The data collection apparatus consists of a mobile device, attached to various camera mounting devices, a mobile device, an iOS app named VirtuCam, as well as a plug-in for Maya. As the user moves the camera in space, the app transfers mobile device motions over a Wi-Fi network, where it could be translated into a Maya camera object. The Maya plug-in sends a video stream of the viewport back to the mobile device display in real-time. This pipeline let the user follow a desired trajectory around a virtual subject, while seeing the results in the "view finder", mimicking a typical shooting workflow in current digital cinematography.

For this prototype, two individuals with different level of expertise in cinematography provided over 500 samples of various camera motions i.e., pan, tilt, dolly as well as their combinations, using various camera mounting equipment and techniques, i.e. handheld, tripod, slider, over-shoulder rigs, etc.

Data processing

To prepare the raw collected data for learning process, several methods have been used to eliminate invalid samples, improve the signal-to-noise ratio, and reduce the effect of data range.

- All samples were transferred from Maya as a sequence of location and orientation values of shape $m \times n \times 6$, where m is the number of samples, n is the number of steps in each sample, x_i, y_i, z_i are the Cartesian coordination of i^{th} target, and u_i, v_i, w_i are the Euler rotation of the i^{th} target.

$$x_i = \begin{bmatrix} x_0 & \cdots & x_{n-1} \\ y_0 & \cdots & y_{n-1} \\ z_0 & \cdots & z_{n-1} \\ u_0 & \cdots & u_{n-1} \\ v_0 & \cdots & v_{n-1} \\ w_0 & \cdots & w_{n-1} \end{bmatrix}$$

- To improve the signal-to-noise ratio and eliminate the high-frequency jittering, a low-pass filter was applied. This filter resulted in a smoother signal for the training phase.
- A common problem in the data set was the existence of extremely large values, potentially from a false sensor reading. To mitigate for this issue, the values were clipped between ± 1000.
- Instead of using the absolute location and rotation values, I opted to use the changes between each pair of consecutive steps.[56] The data was converted into a matrix of shape $m \times (n-1) \times 6$:

---

methods, just to get back and fine-tune the learner, both before and after using the GUI to interactively and visually inspect the results.

[56] Working with Δx instead of x resulted in a smoother motion reconstruction results which will be discussed in more details in the next section. The delta value is a representation of relationship between each time stamp and is more structured than absolute location of each point in space. As each pose in each time stamp is defined based on the delta value from the previous time stamp pose, any error in the delta value will result in a propagated error downstream. This propagated error would affect the loss function and the learning process.

$$\Delta x = \begin{bmatrix} x_1 - x_0 & \dots & x_n - x_{n-1} \\ y_1 - y_0 & \dots & y_n - y_{n-1} \\ z_1 - z_0 & \dots & z_n - z_{n-1} \\ u_1 - u_0 & \dots & u_n - u_{n-1} \\ v_1 - v_0 & \dots & v_n - v_{n-1} \\ w_1 - w_0 & \dots & w_n - w_{n-1} \end{bmatrix}$$

- The data was standardized along each of the 6 degrees of freedom to reduce the negative effects of wide changes in values on the learning process.

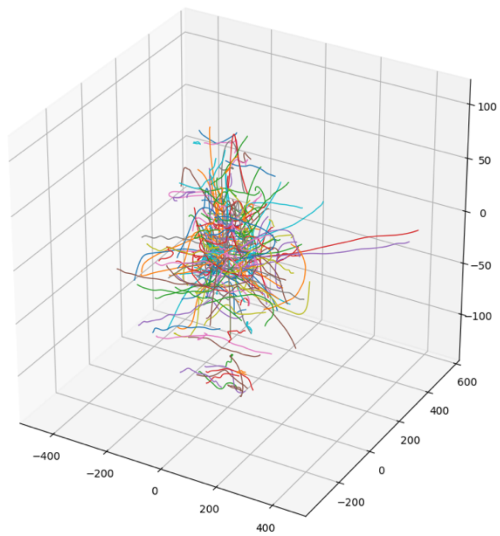

Figure 10- A subset of training sample after preparation and pre-processing.

Learning

The learning phase was focused on training an autoencoder to encode and then decode a given sequence of motions. This process will result in two sub-models, 1) an encoder that maps a high-dimensional data (in this case $1 \times 120 \times 6$) into a lower dimension (i.e.,1×128) latent vector; 2) a decoder network that reconstruct the original motion given its latent vector.

The rationale behind choosing autoencoder was based on its relatively fast and stable training, compared with other generative models such as GANs, as well as the ability to manipulate the latent space vector. For this prototype, I chose AE over VAE due to the simplicity of implementation. As a possible next step, I will implement the model using VAE.

Figure 11- The learning pipeline diagram

During the design phase, several autoencoder models with various type of layers, depth and width of layers, hyper parameters, optimizers were tested. Eventually, I found this architecture the most practical one:

Table 4-Autoencoder details

Element	**Values**	**Note**
Input	Location vector	$(1 \times 120 \times 3)$
	Rotation vector	$(1 \times 120 \times 3)$
	Starting pose	(1×6)
Encoder layers	Fully connected	5, 10, 10 hidden units
	Activation	ReLu
	Drop-out	No
	Regularizer	No
Latent space		(1×128)
Decoder layers	Fully connected	10, 10, 5 hidden units
	Activation	ReLu
	Drop-out	No
	Regularizer	No
Output	Location vector	$(1 \times 120 \times 3)$
	Rotation vector	$(1 \times 120 \times 3)$
	Starting pose	(1×6)
Optimizer	Adam	
	Number of epochs	500
	Batch size	16
	Validation to training ratio	0.15
	Objective function	Mean square error

One important observation in this prototype was the importance of simplicity and preventing the model from learning noises in the data set. While various models with more complex layers such as LSTMs and Convolutional Neural Nets were also tested, they proved to be oversensitive to the small changes in the input data and would generate less desirable results.

Lower latent space dimensions could not efficiently capture the variations in the training data set and performed poorly in reconstruction. However, a lower-dimension latent space is easier to visualize, and the user could interpret each element's effect on the generation process, resulting in better exploring experience.

On the other hand, larger latent spaces open the room for capturing more details of motions and accordingly result in better reconstruction results. However, they pose an extra challenge on the user to comprehend each element's effect and increase the complexity for user interaction, visualization and manipulation.

On a machine with a single 1080Ti GPU the model would converge between 100-200 seconds, using early stopping methods would also reduce this time. The combination of relatively small training data set and the short training duration gives the user the ability to review the results and add/remove samples to steer the learning process.

Upon the completion of the training, all the training sampled were passed to the encoder model to form a matrix of all latent vectors. PCA was applied to this matrix to find 3 dominant elements of the latent vector. The values of these three vectors was used in the GUI to visualize the distribution of motions in a 3d space.

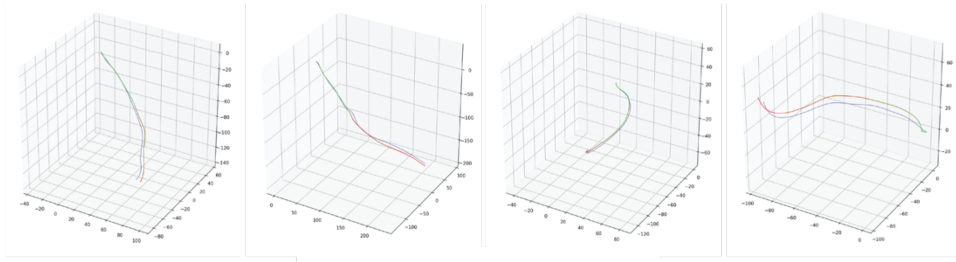

Figure 12- Samples of motions(blue) and reconstructed motions (gradient from green to red). The gradient color signifies the reconstruction error at each pose.

Next step: The learning model can be improved by:

- Testing with different architectures, such as VAE;
- Training two separate models for visualization and reconstruction;
- Migrating from Tensorflow/Keras Python API to Tensorflow for Java Script (TFJS) for a better integration with the GUI.

User Interface

The user interface is solely focused on the review and generation phase; hence it is possible to extend its functionalities to provide an interface for basic neural network design and training.

In this phase, the interface let the user quickly review the 3d distribution of motions and select two arbitrary motions. Then it can interpolate between two motions to find the desired one and directly send it to Maya, either for simulation/visualization or execution by a robot-mounted camera.

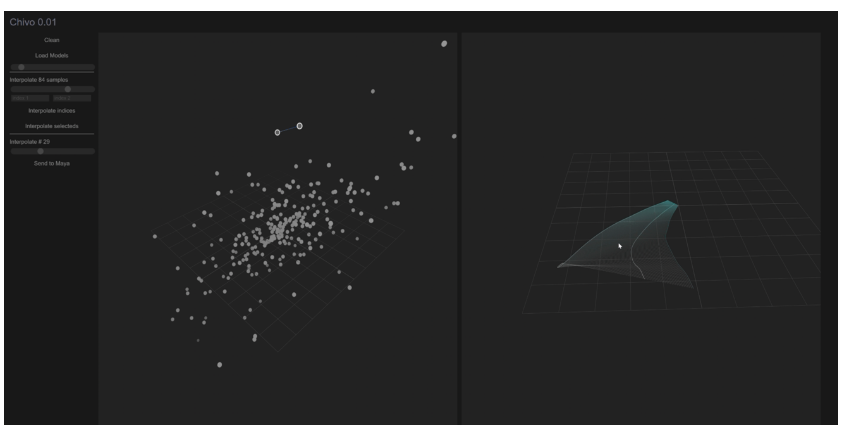

Figure 13- Toolkit interface with 3d distribution of data on the left windows and interpolated motions of two selected motions on the right window.

The GUI can be improved by:

- Integrating the training and user interaction phases by implementing the model in TFJS;
- Introducing weighted interpolation to the interpolation modes;
- Introducing latent vector modification tools;
- Improve visualization techniques for better understanding of the latent space and its effect + version history.

Next Steps

A critical next step in the protype development is to conduct a series of user experience studies to test the practicality of the framework in action. I will design the details of user studies and benchmarks as the research proceeds.

The prototype is still under development and the results will be submitted for SIGGRAPH 2020, which is due in mid-January 2020.

Further Toolkit Development

A challenge in the process of developing the first prototype was the lack of a multipurpose platform to serve as the host for the toolkit. I developed the ML backend in Python and bundle it with a Web-app using Django framework. Using a custom-made interface requires significant resource allocation to developing various basic functionalities to support data pipelines integrations, GUI elements, visualization, user interactions and leaves less room to focus on the toolkit core functionalities.

Based on my experience in this phase, it might be more practical to host this toolkit on an off-the-shelf platform such as Runway which can provide us with the infrastructure—version control, sharing, cloud computing, in addition to GUI elements and multi-platform support. I will make decision on the platform after further investigation of Runway current version and its capabilities and discussion with the core developers of their team.