Architecture, Design and Conservation
Danish Portal for Artistic and Scientific Research

Aarhus School of Architecture // Design School Kolding // Royal Danish Academy

Re/Learning the Wheel

Rossi, Gabriella; Nicholas, Paul

Published in:
ACADIA // 2018: Recalibration. On imprecisionand infidelity

Publication date:
2019

Document Version:
Publisher's PDF, also known as Version of record

Link to publication

Citation for pulished version (APA):
Rossi, G., & Nicholas, P. (2019). Re/Learning the Wheel: Methods to Utilize Neural Networks as Design Tools for Doubly Curved Metal Surfaces. In *ACADIA // 2018: Recalibration. On imprecisionand infidelity: Proceedings of the 38th Annual Conference of the Association for Computer Aided Design in Architecture* (pp. 146-155)

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

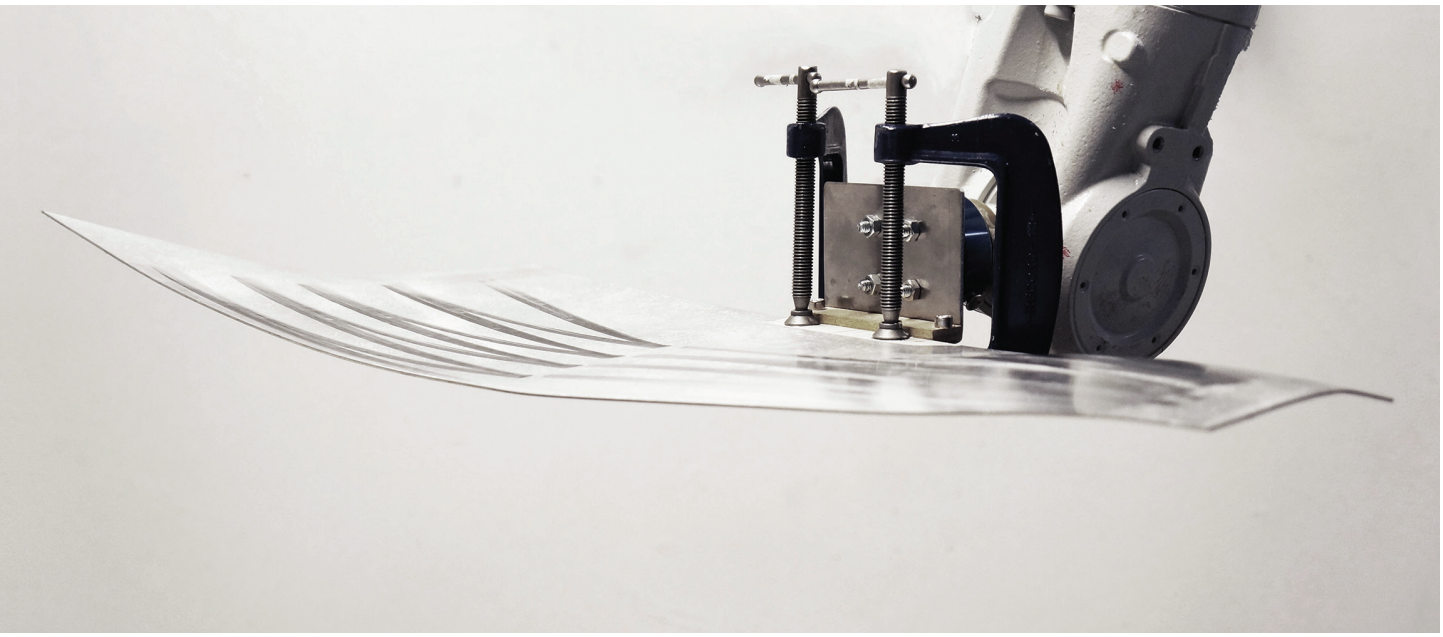
Download date: 25. Apr. 2024

ARKITEKTSKOLEN AARHUS
Designskolen Kolding
Det Kongelige Akademi

Re/Learning the Wheel

Gabriella Rossi
Paul Nicholas
CITA/Royal Danish Academy of Fine Arts

Methods to Utilize Neural Networks as Design Tools for Doubly Curved Metal Surfaces



1

ABSTRACT

This paper introduces concepts and computational methodologies for utilizing neural networks as design tools for architecture and demonstrates their application in the making of doubly curved metal surfaces using a contemporary version of the English Wheel. The research adopts an interdisciplinary approach to develop a novel method to model complex geometric features using computational models that originate from the field of computer vision.

The paper contextualizes the approach with respect to the current state of the art of the usage of artificial neural networks both in architecture and beyond. It illustrates the cyber physical system that is at the core of this research, with a focus on the employed neural network–based computational method. Finally, the paper discusses the repercussions of these design tools on the contemporary design paradigm.

1 The method creates sinuous doubly curved metal panels that are characterized by a manufacturing marking unique to this process. The location, orientation, and density of the tracking pattern dictate the final geometry of the panel.

2 Like a craftsman, the robot adapts its motion according to the shape of the piece it is holding in order to roll it along the English wheel placed in front of it.

INTRODUCTION

The contemporary architectural design paradigm is a two-step linear process: design through drawing, then the pushing of numerical data into manufacturing. The workflow is segmented across the industry between architects and fabrication specialists (Callicott 2005). This linear design-to-production, file-to-factory means of organizing production represents a missed opportunity in terms of design possibilities: we can only build what we can draw a priori using standardized notations (Mitchell 2001; Carpo 2011). As a result we are leaving behind "crafty" fabrication processes that resist numeration because of their complex nature, even though they can be productive methods of creating certain architectural forms and expressions.

This focus on predefined tolerance and control is contrasted by a more tolerant design-through-making iterative paradigm, and it marks the main schism between modern construction and craft. Craftsmen's cognitive system is based on continuous feedback between body, brain, tool, and material to create an artefact (Sharif and Gentry 2015). They requires years of training and skill honing; therefore, utilizing their expertise on an architectural scale makes it very labor intensive and expensive.

A digital design-through-making approach, linking a robot with artificial intelligence, explores the relationship between the material and its behavior during the manufacturing process as a design possibility. This requires a fabrication model that is continuously evolving and able to account for and learn from new factors emerging during the manufacturing process (Sharif and Gentry 2015).

Artificial neural networks (ANN) lend themselves to this concept. Inspired by the biological aspects of learning, they present a genetically encoded part (their architecture, optimizer and loss function for instance) and a variable part (weights and biases) that changes through trial and error using a reward system (Ashby 1954). Similar to human learning, machine learning algorithms aim to change these variables in order to define a better relationship between input and outputs. They recalibrate at every iteration, without having the final configuration explicitly preprogrammed. They converge towards an approximation of the relationship that is deemed good enough, yet not error free.

This research explores how a transfer of craft principles, namely the fabrication of doubly curved metal surfaces using the English wheel, as well as a computational model, namely convolutional neural networks, into an architectural design environment, Rhino/Grasshopper, could provide an interesting framework for the making of novel architectural elements. This is achieved through the implementation of an adaptive cyber-physical system as a method to model this rich and complex fabrication method. Much of the challenge comes from adapting a computational model that originates in the field of computer vision into an architectural design workflow (Figure 2).

2

The paper is organized as follows: section one introduces the state of the art of deep neural network applications in the architectural field and of convolutional neural networks in the fields of computer vision and medical image processing. It highlights productive territories of interdisciplinary cross-pollination. Section two describes our implementation of the cyber-physical system with a focus on the computational workflow in order to create a novel design tool that utilizes these neural networks.

BACKGROUND

ANNs have gained popularity in recent years due to advances in hardware that are able to handle larger computations, as well as the availability of large open source datasets able to be utilized for the training of new models (Goodfellow et al. 2016). They have quickly spread to a large number of fields. In the following section we will focus on examples of feedforward neural networks, that is,

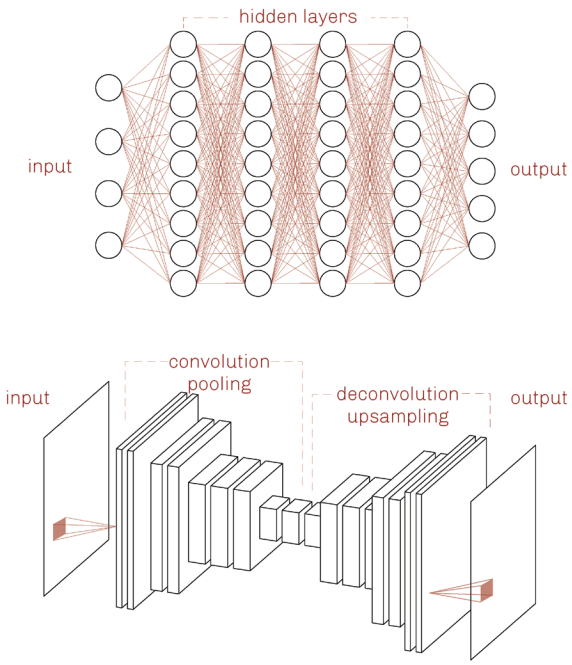
those that present a directional graph where backpropagation is used to calibrate the nonlinear relationship between input and output. We will compare the network architecture and the data acquisition method with the obtained results (Figure 3).

Neural Networks in Architectural Applications

Within the architectural robotic fabrication context, regression is the most used machine learning technique, since it is able to predict specific parameter values given a certain dataset of training examples. For problems of limited complexity, where only one parameter is correlated to the other, regular data-mining techniques are used. For instance, Smigielska uses polynomial regression to predict the springback of metal rods that undergo robotic bending (2018). Cheng and Hou use Spearman's rank correlation in order to evaluate the stability of a stacked assembly of branches based on their swept area (2016). When the number of parameters to be correlated increases, regression via ANN is implemented. Similarly to how ANNs are used in the manufacturing industry (Al Zubaidi et al. 2011; Kashid and Kumar 2011), their usage in architecture attempts to optimize digital fabrication parameters. The topological complexity of the network is dictated by the complexity of the task to be performed.

An example of this approach is the work of Brugnaro and Hanna (2017). Their implemented network deals with multiple features of chiseling wood, namely tool/surface angle, tool/grain angle, force feedback, feed rate, cut length, and cut depth. The dataset was recorded using motion tracking cameras that tracked the movements of a craftsman using a chisel and streamed it into Grasshopper, recording 1500 entries. A deep neural network with a 5-30-1 neuron architecture is implemented to predict one parameter at time based on the other five. This has been tested in predicting tool angle variation with a 2.12° error, and cut depth with a 0.31 mm error.

Another example of regression neural networks—and perhaps the most complex in terms of features—is the work of CITA on the A Bridge Too Far project, where a deep neural network has been implemented in order to predict the springback of incrementally formed metal panels. In order to overcome the limited amount of data available, every scanned panel was rotated and mirrored in order to obtain 4 panels, and it was then divided into 5 x 5 cm samples that each became a 100 feature tensor, to which two more features were added: the distance of the patch from the panel edge, as well as its distance from the frame edge. This data augmentation technique yielded over 45,000 samples, that were trained using a 150-80-30-1



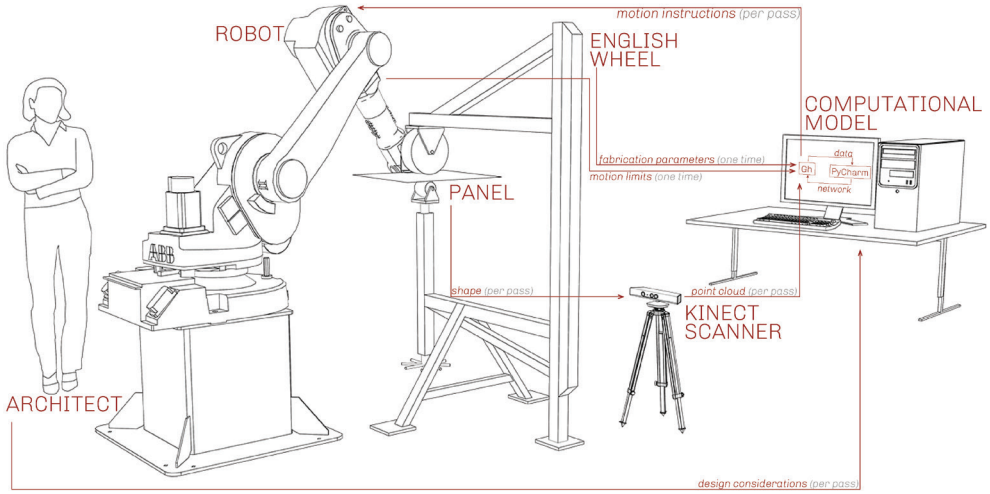
3

ANN. The network was able to predict the fabricated geometry from an input mesh with an error rate of less than 3 mm, although the inverse prediction (fabricated geometry to input geometry) was less precise (Zwierzycki, Nicholas, and Thomsen 2018).

Differently from the above, the façade of the Princeton Embodied Computation Lab by The Living/Autodesk provides an example of utilizing machine learning algorithms not for fabrication parameter optimization, but for geometric localization of a CNC sandblasting toolpath. The network had a task of localizing knots in reclaimed scaffolding wooden planks. As per general practice in image recognition, a convolutional neural network was used to classify square segments of the plank as knot or not. This technique, usually used in the milling industry to get rid of knot defects (Norlander, Grahn, and Maki 2015), is actually used to accentuate the aesthetic expression of the knot after being subjected to sandblasting. The training dataset was obtained by crowdsourcing through a web app, which presented a photo of a segment for users to classify as either knot or not. This allowed the network to learn to classify 2,000 boards, while also generating a heat map of where the knot was, which was translated into a CNC toolpath for the sandblasting machine (Nagy 2017).

Neural Networks Outside of Architecture

This last project is an example of the broader state of the art of neural networks, presenting interesting models to be applied into the architectural field. Convolutional neural networks (CNN), for instance, have huge potential



4

3 Deep neural networks *(top)* handle one-dimensional arrays of data through their neurons and a multitude of connections, whereas fully convolutional networks *(bottom)* handle two-dimensional arrays of data though their localized neuron connections.

4 Cyber-physical setup for Robotic English Wheeling, creating a link between physical making, digital representation, and neural network training.

in geometric applications: differently from dense deep networks (multilayer perceptrons used in the above-mentioned cases), CNN architecture is characterized by its ability to extract local features by restricting the receptive field of the hidden neurons to be limited to a small neighborhood of the previous layer. This local receptive field convolutes over the data using a preset kernel and it presents a shared weight matrix. This means that variables that are spatially near one another will be highly correlated, giving the network the ability to extract spatial features with partial indifference to scaling or shifting (LeCun et al. 1998). This sparse connectivity and spatial subsampling method makes CNNs powerful tools to process data that has a known grid-like topology of any size (Goodfellow et al. 2016), such as photos, which explains their proliferation in the field of computer vision and object recognition.

State-of-the-art CNNs are usually benchmarked according to their accuracy in classifying standardized datasets such as the MNIST (handwritten numbers from 0 to 9) or the CIFAR10 (labeled images of different things). The input is the image, and the CNN identifies the local features and outputs the classification label with the highest probability. Fully convolutional networks (FCN) push this concept further by outputting a dense pixel-by-pixel label classification, called semantic segmentation. To do so, they replace the final fully connected layer of a normal CNN with more convolutional layers (deconvolutions) in an hourglass fashion, which are able to upsample the predictions to cover every pixel (Long, Shelhamer, and Darrell 2015). CNNs also offer the possibility to convolute in three dimensions: VoxNet subsamples a pointcloud into voxel neighborhoods, which enables it to classify different pointclouds of objects into their corresponding labels (Maturana and Scherer 2015).

Finally, another field that is quickly developing interest in CNNs, also because of its geometric aspects, is medical image computing. Networks are trained to detect anomalies and diseases in 2D imagery (Li et al. 2014), as well convolute over a stack of 3D MRI imagery. In so doing, they not only relate spatial features on the same slice, but also correlate them with the same position in the previous and next slices in order, for example, to produce accurate segmentation of brain lesions (Kamnitsas et al. 2016)

METHOD

In this section we introduce the larger context of research for which the neural network–based computational method was developed. The research investigates the possibilities of a contemporary approach to fabricating doubly curved metal surfaces using the English wheel as a hybrid digital craft. Understanding the relationship between the material and its behavior during the fabrication process was the key to establishing design control over geometries made with the English wheel.

Cyber-Physical Setup for Robotic English Wheeling

The cyber-physical setup (Figure 4) is composed of a Dinosaurier English wheel placed in front of an ABB IRB1600 robot arm, with a Kinect scanner ensuring the feedback between the analogue and the digital world. The robot is seen not only as a manufacturing tool, but also as a first-person design agent aware of the material impacts and causal effects of fabrication actions, in the same way as a craftsman would be; through interaction, constructive memory, and situatedness (Gero 2017). It proposes designing an ever-growing brain that acquires knowledge at every iteration, and that could develop a digital intuition to Robotic English Wheeling.

The underlying mechanical principle of the English wheel is technologically simple: it is composed of two wheels, a flat one and a crowned one, between which a sheet of metal is moved back and forth. This stretches the material and forces it out of plane, which creates curvature. In our setup, a flat sheet is clamped onto the holding end-effector, and the robot pushes the sheet back and forth in a zigzag fashion—its tracking pattern—on the English wheel placed before it. The motion is not planar, in that a minimal slope is introduced in the toolpath in order to allow the robot to dissipate the torque on axis 4, and thus be able to work the sheet in the same way a craftsman would. The robotic arm is controlled from the Robots plugin for Grasshopper, which generates quaternion motion based on a calibrated file representing the workshop with several reference points: the center point of the crowned wheel, the edge of the holding end-effector, the scanning position, and the dataset generation for the neural network's learning position. The relationship between these positions is an exercise of plane reorientations. After forming, the robot moves the sheet into a scanning position, where the Kinect transfers the pointcloud to Grasshopper using Firefly. The infrared scanning capacity of the Kinect avoids the creation of occlusions and noise due to light reflections over the metal surface while delivering a pointcloud with an acceptable tolerance. This guarantees a realtime feedback loop between the physical prototypes and their digital representation. This is necessary to enable the robot to do multiple passes over the same piece, since after the first pass is done at high pressure, the sheet is no longer flat and the motion toolpath has to be adapted in order to avoid physical damage.

The main difference between the human and robotic craft process is the robot's ability to apply more pressure than a human (0.08 mm gap between the wheels vs 0.2 mm gap), thus it can produce bespoke pieces in fewer passes and less time. This increased forming pressure leaves behind a tracking pattern trail unique to Robotic English Wheeling, which becomes an additional aesthetic element of design and surface expression, alongside its geometry, for the doubly curved sheet (Figure 1).

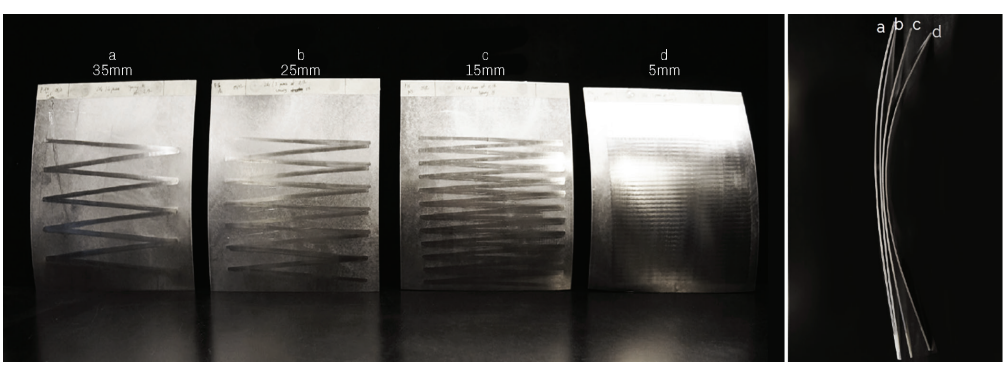
The Need for a Convolutional Neural Network

A neural network was considered as a solution to model the fabrication system, although the network had to go beyond the state of the art analyzed above: the geometries in question are doubly curved and cannot be encoded as a heightfield, as were the incrementally formed panels of A Bridge Too Far. In fact, every pixel should be encoded with more than one feature that would convey the 3D shape of the panel. Moreover, they had reported that the regression-based ANN produced significantly more error in the areas with a large difference in the slope gradient (Zwierzycki, Nicholas, and Thomsen 2018). This was due to the loss of the spatial data structure during the data processing into flat tensors. Both of these issues could be solved by implementing a CNN/FCN.

Data Acquisition and Preprocessing

As training dataset, we chose to use an ensemble of six 250 x 250 mm, 1.5 mm thick aluminum panels that were formed into synclastic curvatures using the robotic English wheel in one pass (Figure 5). The Kinect pointclouds were cropped and cleaned, and the surfaces representing the formed panel was reconstructed. In order to augment the dataset, every surface was rotated 90,180, and 270 degrees. The surfaces were then reparametrized from 0 to 1. Each was then sampled into a 64 x 64 point grid (approximately a point every 4 mm). The choice of the number 64 was to simplify upcoming network architecture topology, since CNNs are known to work better with base 2 numbers.

The usage of a CNN required us to translate the geometry into two-dimensional tensors to start with. The tracking pattern data was encoded by overlaying the toolpath polyline curve onto the surface and intersecting it with the 64 x 64 sampling grid to generate a binary map (Figure 6). Since all machining operations were done on one side of the sheets, there was no need for a third dimension to the

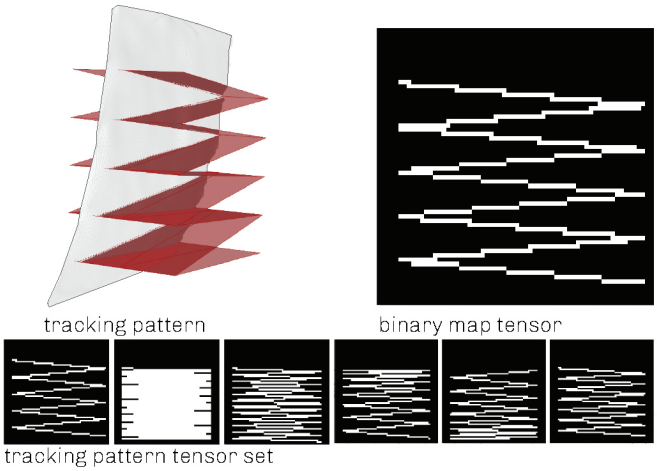


5

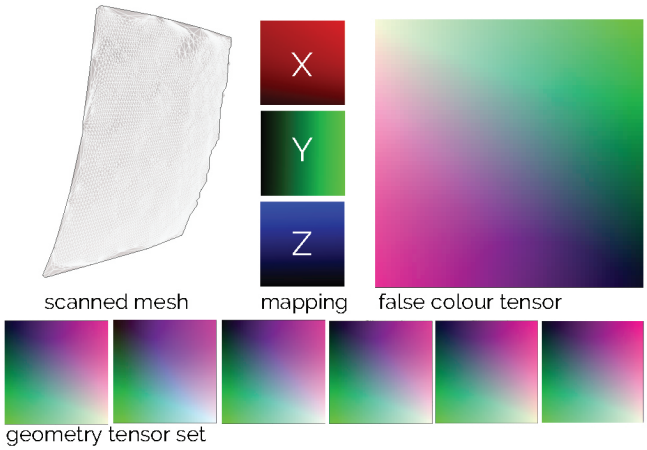
5 Four out of the six pieces of the selected dataset: different spacing of the tracking pattern producing different curvatures of the sheet

6 Tracking pattern training tensor set composed of binary maps

7 Geometry training tensor set composed of false color maps



6



7

tensor. The geometry data required more of a workaround to encode. An initial attempt contoured the formed surface using the initial flat sheet orientation, yielding a binary map of contour lines. What seemed to be a good representation of both depth and orientation of the surface did not yield good result in the learning phase. The second more robust method uses the point coordinates themselves to generate a false color bitmap. By mapping the distribution of the x coordinates for the 64 points of every panel in red, the y coordinates in green, and the z coordinates in blue, every sample point became a three-dimensional tensor and the whole panel became an RGB image (Figure 7).

A second data augmentation workaround was implemented to overcome the limited number of panels: each of the now 24 panels was cropped into 625 16 x 16 pixel subsamples. This was done by moving a cropping frame pixel by pixel along the x and y directions. Finally, as is common practice in neural network development, the dataset was split into train and test sets, with 5 of the original panels to train the network (12,500 tensors in total) and 1 original panel to test its performance (2,500 tensors in total).

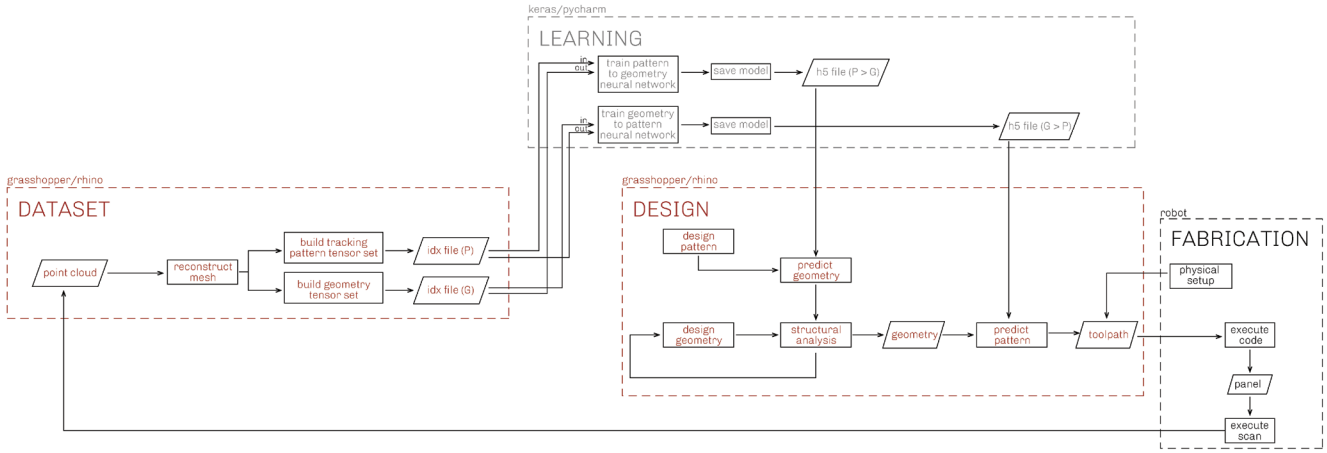
Learning

The datasets were wrapped into IDX files using Owl for Grasshopper. This file type is an efficient format to store tensors as multidimensional arrays. They are taken into PyCharm (an integrated development environment for Python) for neural network training using Keras for Python running on a TensorFlow backend. First we trained the network to predict the tracking pattern necessary to obtain a certain 3D geometry. A fully convolutional network was implemented to make a dense pixel by pixel prediction. The architecture was similar to a convolutional autoencoder: the first half of the network alternates between three convolutional layers and maxpool layers with 8, 16, and 32 filters, respectively, with a kernel of 7 x 7 pixels and using a ReLU activation function. It downsamples the image, then the second half mirrors the first half and upsamples the image back to the original size. Finally, a last convolutional layer produces one channel of results, the tracking pattern binary tracking pattern map, using a sigmoid function.

Teaching the network how to predict the geometry from a given tracking pattern kept the same convolutional autoencoder architecture; however, tweaking the filter and kernel was necessary. The network uses 256, 128, and 64 filters, respectively, with a decreasing kernel of 7 x 7, 5 x 5, and 3 x 3. The last convolutional layer outputs three channels of results, which are then decoded into RGB bitmaps back in the Grasshopper environment.

Design Tool

After training the neural network for 2,000 iterations, the configuration of weights and biases matrices needs to be frozen. They are saved as h5 files, which are adequate for these types of multidimensional arrays to be loaded into PyCharm whenever an evaluation in needed, or to continue training. This needs a separate Python script that calls the saved network and performs a prediction. This script can be called directly from within Grasshopper to run in the background as a Python process. The output predictions are saved as IDX files and loaded into Grasshopper for visualization. This allows the designer to make use of the trained models in both directions: as a way to design a surface and observe the resulting imprinted pattern, which is the traditional design-to-fabrication sequence, but also in a more crafty way of designing the desired fabrication marks and observing their effect on the geometry (Figure 8).



8

8 Flowchart of the Robotic English Wheeling process: a fabrication model that is flexible, adaptive, and in constant recalibration. It learns by incorporating the newly made panels back into the knowledge dataset. Data flows between the neural network training environment and the design environment.

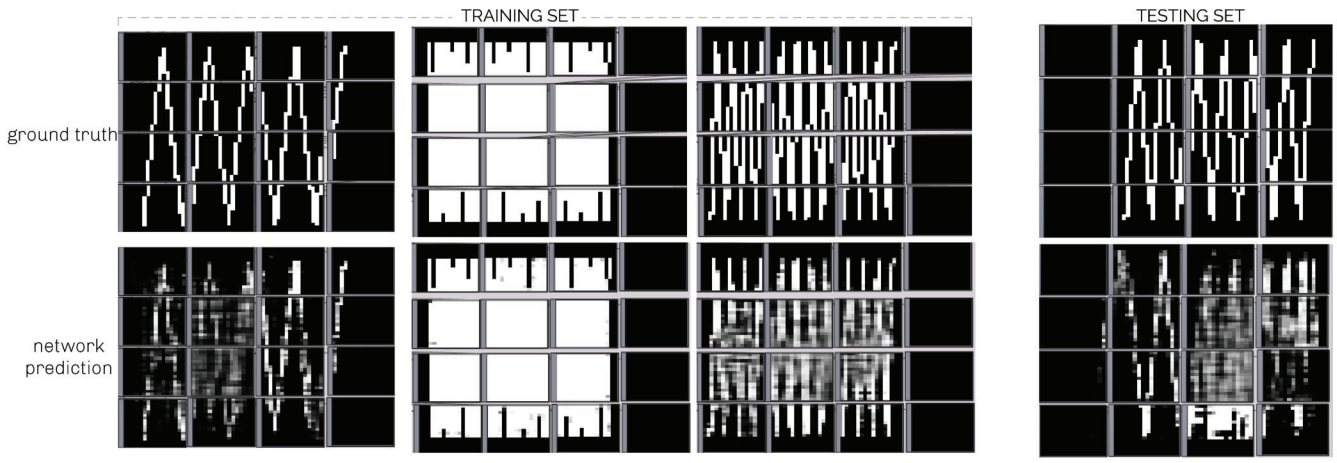
RESULTS & DISCUSSION

This computational workflow exploration has been successful in translating a doubly curved surface and its tracking pattern into a mathematical tensor that is usable by an FCN. The first network that translates geometry to tracking pattern has a precision rate of 94% over the training dataset and a precision rate of 75% over the test dataset (Figure 9). It is suspected that this decrease in performance between the two is the result of slight overfitting due to the limited diversity of training samples. However, considering that the toolpath has to be post-processed in Grasshopper regardless, in order to ensure fabrication feasibility and collision avoidance of the robot using the wheel, this is acceptable in our case. On the other hand the second network that translates tracking pattern to geometry has a precision rate of 85% over the training dataset and a precision rate of 65% over the test dataset. This should be improved in further work, perhaps by adding more features or accentuating the differences in the false color mapping (Figure 10).

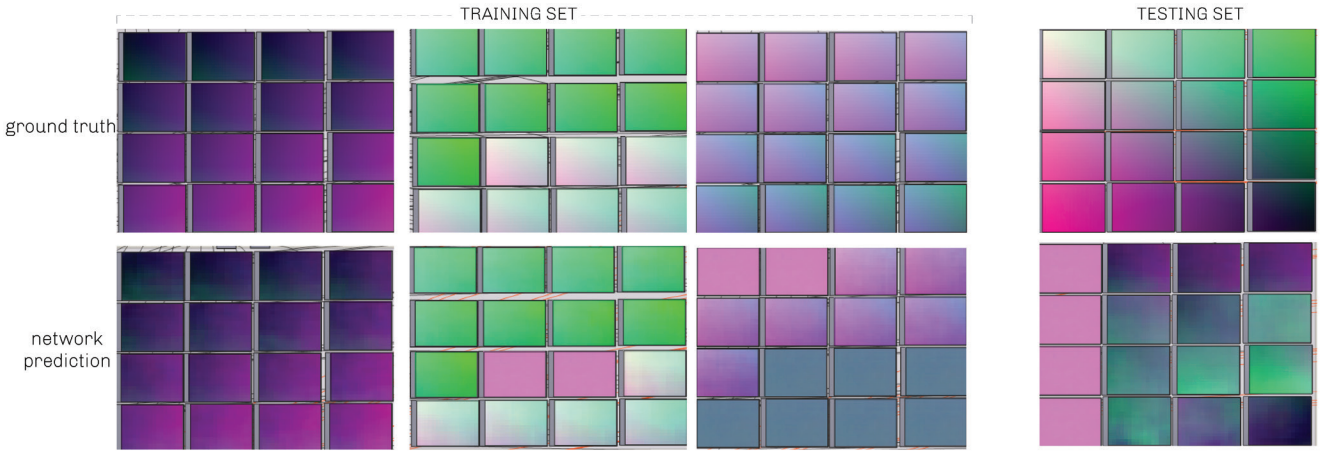
This approach allows for the use of complex neural networks that originate in the field of computer vision as a design tool for architectural purposes. This tool puts the designer at the center of the design process: capable of shaping the process and its results by deciding which datasets are to be used, and thereby controlling the knowledge base of the neural network, since it can only predict what it has been allowed to learn. Since the datasets are composed of real physical panels, the decision making of the designer is based on a qualitative evaluation of their design and geometric quality, thus reconciling design and making in the architectural ideation process, bringing it closer to craft practices in a powered-up digital version.

In typical non-architectural neural network applications, the data is usually preexisting and has already undergone cleaning processes. This approach can only have very limited applications within architectural workflows, which deal more often with the creation of new data. In "live" situations such as fabrication, a cyber-physical approach supports both the harvesting of clean data and the making of actions based on neural networks. Considerations exist around the cleaning and representation of data in this approach as much as they do in traditional approaches, particularly around issues of noise. However, if we are to further engage with the physical and production aspects of architecture, cyber-physical systems represent a best way forward to allow architects to create their own datasets relevant for their particular task. Future research will seek to build a more varied dataset comprising a larger variety of surfaces—synclastic and anticlastic—with tracking patterns machined on both sides of the panels. This will perhaps require us to shift the network topology from 2D convolutions to 3D convolutions in order to cope with the possibility of an extra dimension of the data array.

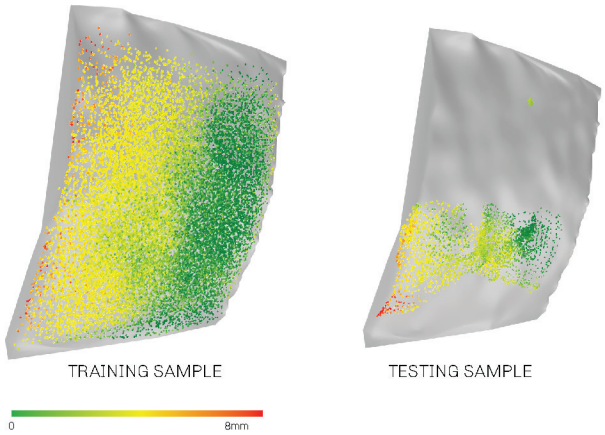
This approach also questions the idea of tolerance and imprecision in the design process: a neural network–based tool is by far less precise than a specialized FEA analysis programmed by a material specialist. However, we suggest that the added value of having such a tool within a design environment such as Grasshopper certainly balances out its imperfect precision (Figure 11). It allows the designer to have more control over the making of architectural components outside of a closed specialized industrial environment, and thus trades off tolerance for design exploration freedom. This is why it is of great importance that architects acquire a digital literacy with regards to advances



9



10



11

9 Results of the geometry-to-tracking pattern neural network over samples

10 Results of the tracking pattern-to-geometry neural network over samples

11 Prediction of the points position over samples using the tracking pattern-to-geometry network. This illustrates the sensitivity of the false color images in translating back into geometric position of points.

in neural network techniques that are relevant to architectural geometries, which can act as an augmentation to their knowledge. They should also seek collaborations with computer science specialists for better interdisciplinary cross-pollination and the development of tools that are best suited for architects' needs. The development of tools such as Owl or LunchBox ML are visionary steps hinting at the future democratization of neural networks. However, the danger of these plugins' closed solvers is that they only allow for certain types of simple networks. The proposed workflow, using Keras and Owl to package the data to and from Grasshopper, has been deemed simple enough to not require years of computer science study, but free enough to let the designer shape the network, and not vice versa.

CONCLUSIONS

This paper has presented a conceptual and methodological framework of implementing convolutional neural networks as a design tool for doubly curved surfaces made with the English wheel. An initial method to translate complex architectural geometry and fabrication features into mathematical matrices understandable by a neural network opens the door for further implementation of CNN as design tools in architecture. The project more broadly suggests the rethinking of the file-to-factory design paradigm in favor of a more iterative design-through-making process.

Neural networks applications are certainly growing to become popular in more and more industries, computational design included. However, whether they can bring added value to design processes, fabrication processes, or both will depend on how well architects understand and interface with these tools—in short, whether they will be able to appropriate neural networks into making them their own, either through novel methods of acquiring and representing datasets or in shaping the architecture of the networks themselves. After all, a neural network is nothing but an element of design.

ACKNOWLEDGEMENTS

This paper presented the research development of the master's thesis by Gabriella Rossi, supervised by Prof. Paul Nicholas within the CITAstudio: Computation in Architecture Master Program at the Royal Danish Academy of Fine Arts, Copenhagen. The authors would like to thank Mateusz Zwierzycki for further developing the Owl plugin for Grasshopper.

REFERENCES

Al-Zubaidi, Salah, Jaharah A. Ghani, and Che H.C. Haron. 2011. "Application of ANN in Milling Process: A Review." *Modelling and Simulation in Engineering* 2011: 696275.

Ashby, W. Ross. 1954. *Design for a Brain*. New York: John Wiley & Sons.

Brugnaro, Giulio, and Sean Hanna. 2017. "Adaptive Robotic Training Methods for Subtractive Manufacturing." In *Disciplines & Disruption: Proceedings of the 37th Annual Conference of the Association for Computer Aided Design in Architecture*, edited by Takehiko Nagakura, Skylar Tibbits, Mariana Ibanez, and Caitlin Mueller, 164–9. Cambridge, MA: ACADIA.

Callicott, Nick. 2005. "Adaptive Architectural Design." *Architectural Design* 75 (4): 66–9.

Carpo, Mario. 2011. *The Alphabet And The Algorithm*. Cambridge, MA: MIT Press.

Cheng, Chi-Li, and June-Hao Hou. 2016. "Biomimetic Robotic Construction Process: An Approach for Adapting Mass Irregular-Shaped Natural Materials." In *Complexity & Simplicity: Proceedings of the 34th eCAADe Conference* vol. 1, edited by Aulikki Herneoja, Toni Österlund, and Piia Markkanen, 133–42. Oulu, Finland: eCAADe.

Gero, John. 2017. "Cognitive Design Computing" In *ShoCK!: Sharing Computational Knowledge!—Proceedings of the 35th eCAADe Conference*, vol. 1, edited by A. Fioravanti, S. Cursi, S. Elahmar, S. Gargaro, G. Loffreda, G. Novembri, and A. Trento, 37–40. Rome: eCAADe.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. Cambridge, MA: MIT Press.

Kamnitsas, Konstantinos, Christian Ledig, Virginia FJ Newcombe, Joanna P. Simpson, Andrew D. Kane, David K. Menon, Daniel Rueckert, and Ben Glocker. 2017. "Efficient Multi-Scale 3D CNN with Fully Connected CRF for Accurate Brain Lesion Segmentation." *Medical Image Analysis* 36: 61–78.

Kashid, Sachin, and Shailendra Kumar. 2012. "Applications of Artificial Neural Network to Sheet Metal Work: A Review." *American Journal of Intelligent Systems* 2 (7): 168–76.

LeCun, Yann, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. "Gradient-based Learning Applied to Document Recognition." *Proceedings of the IEEE* 86 (11): 2278–324.

Li, Qing, Weidong Cai, Xiaogang Wang, Yun Zhou, David Dagan Feng, and Mei Chen. 2014. "Medical Image Classification with Convolutional Neural Network." In *Proceedings of the 13th International Conference on Control, Automation, Robotics & Vision*, 844–8. Singapore: ICARCV.

Long, Jonathan, Evan Shelhamer, and Trevor Darrell. 2015. "Fully Convolutional Networks for Semantic Segmentation." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3431–40. Boston: CVPR.

Maturana, Daniel, and Sebastian Scherer. 2015. "Voxnet: A 3D Convolutional Neural Network for Real-time Object Recognition." In *Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 922–8. Hamburg: IROS.

Mitchell, William. 2001. "Roll Over Euclid: How Frank Gehry Designs and Builds." In *Frank Gehry, Architect*, edited by J. Fiona Ragheb, 352–63. New York: Guggenheim Museum Publications.

Nagy, Danil. 2017. "Princeton School of Architecture." http://danilnagy.com/embodied-computation-lab

Norlander, Rickard, Josef Grahn, and Atsuto Maki. 2015. "Wooden Knot Detection using ConvNet Transfer Learning." In *Image Analysis: Proceedings of the 19th Scandinavian Conference on Image Analysis*, edited by Rasmus R. Paulsen and Kim S. Pedersen, 263–74. Cham: Springer.

Sharif, Shani, and T. Russel Gentry. 2015. "Design Cognition Shift from Craftsman to Digital Maker." In *Emerging Experience in Past, Present and Future of Digital Architecture, Proceedings of the 20th International Conference of the Association for Computer-Aided Architectural Design Research in Asia*, edited by Y. Ikeda, C.M. Herr, D. Holzer, S. Kaijima, M.J. Kim, and M.A. Schnabel, 683–92. Daegu, Korea: CAADRIA.

Smigielska, Maria. 2018. "Application of Machine Learning within the Integrative Design and Fabrication of Robotic Rod Bending Processes." In *Humanizing Digital Reality: Design Modelling Symposium Paris 2017*, edited by K. De Rycke, C. Gengnagel, O. Baverel, J. Burry, C. Mueller, M. Nguyen, P. Rahm, M.R. Thomsen, 523–36. Singapore: Springer.

Zwierzycki, Mateusz, Paul Nicholas, and Mette Ramsgaard Thomsen. 2018. "Localised and Learnt Applications of Machine Learning for Robotic Incremental Sheet Forming." In *Humanizing Digital Reality, Design Modelling Symposium Paris 2017*, edited by K. De Rycke, C. Gengnagel, O. Baverel, J. Burry, C. Mueller, M. Nguyen, P. Rahm, M.R. Thomsen, 373–82. Singapore: Springer.

IMAGE CREDITS

Figures 1–11: © Gabriella Rossi, CITA/KADK, 2018

Gabriella Rossi is a trained architect interested in applying computational processes and robotic fabrication to architecture. Her current interest lies in applying Machine Learning techniques to rethink architectural design and fabrication workflows. Gabriella has assisted computational research projects at CITA and Politecnico di Milano. She has completed her M.A. with Honours from the CITAStudio: Computation in Architecture program at the Royal Danish Academy of Fine Arts in Copenhagen, and holds an Architecture B.Sc. Cum Laude from Politecnico di Milano. She is member of the Danish Association of Architects.

Paul Nicholas holds a PhD in Architecture from RMIT University, Melbourne, Australia. He is an Associate Professor at the Centre for Information Technology and Architecture (CITA), KADK Copenhagen Denmark and head of the CITAstudio: Masters in Computation international masters' program.Paul has previously practiced with Arup Engineers from 2005 and AECOM/Edaw from 2009. Paul's research interests lie in the development of innovative computational approaches that extend architecture's scope for design by establishing new bridges between design, structure, and materiality. His recent research explores sensor-enabled robotic fabrication, multiscale modeling, and the idea that designed materials such as composites necessitate new relationships between material, representation, simulation, and making.