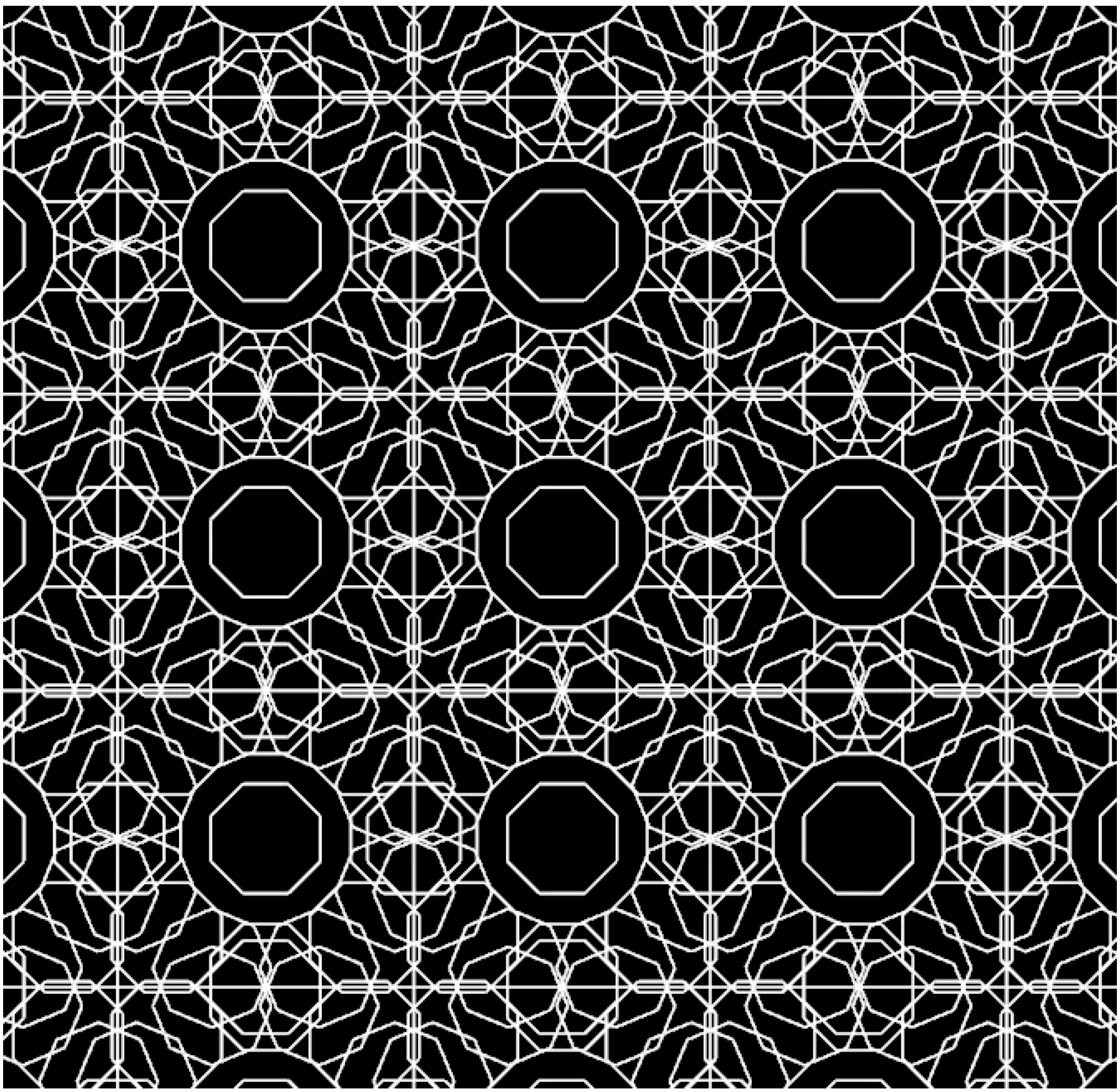
Maths to Magic and Visual Wizardry: Exploring the Procedural Creation of Islamic Geometrical Patterns in Houdini

Georgia Sweeny
Supervised by Oleg Fryazinov
NCCA
Bournemouth University



1. *Abstract*

Visual artists have access to increasingly powerful software to aid in creation of 2D and 3D art and design. However, these programmes can be complex and take years to master, often requiring an existing foundation in 2D/3D tools or coding languages in order to use. Islamic geometric patterns display the intersection of mathematics and art. Previous research has demonstrated that these designs can be recreated with code and applied visual shape grammar using 3D tools like Maya.

This paper presents an accessible proof of concept tool, in the form of a digital asset, for Houdini that allows the user to intuitively generate Islamic geometrical patterns. Houdini's procedural nature allows artists to visualise the end result as they make changes instead of waiting for the final output to be recalculated.

Through implementing and adapting an existing set of shape grammar rules used to create islamic geometric motifs into my Houdini network, I was able to develop a successful working tool. The process outlined in this paper enables a user with no prior knowledge of Houdini software to create unique islamic geometric styled patterns and motifs through adjustable parameters in the user interface.

Key Words: Islamic Geometric Pattern, Shape Grammar, Motif, Procedural, Parameters, Tessellation, Tiling, Digital Asset, Network, User Interface

Abbreviations

Islamic Geometric Pattern - IGP
Shape Grammar Rules - SGR
User Interface - UI
Two Dimensional - 2D
Three Dimensional - 3D

Contents

2. List of Figures

Figure 1.

16th century Sevillian groin tiles in the Sala de los Abencerrajes

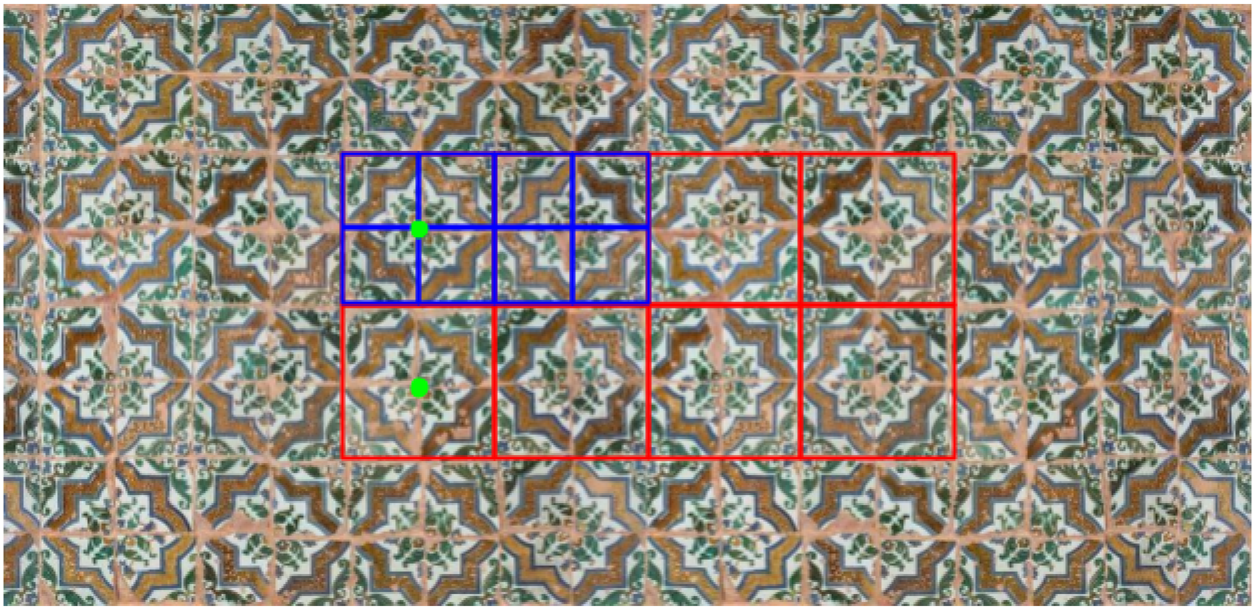


Figure 2.

Compass and ruler construction of an Islamic geometric pattern (Jowers, I. *et al.*, 2010)

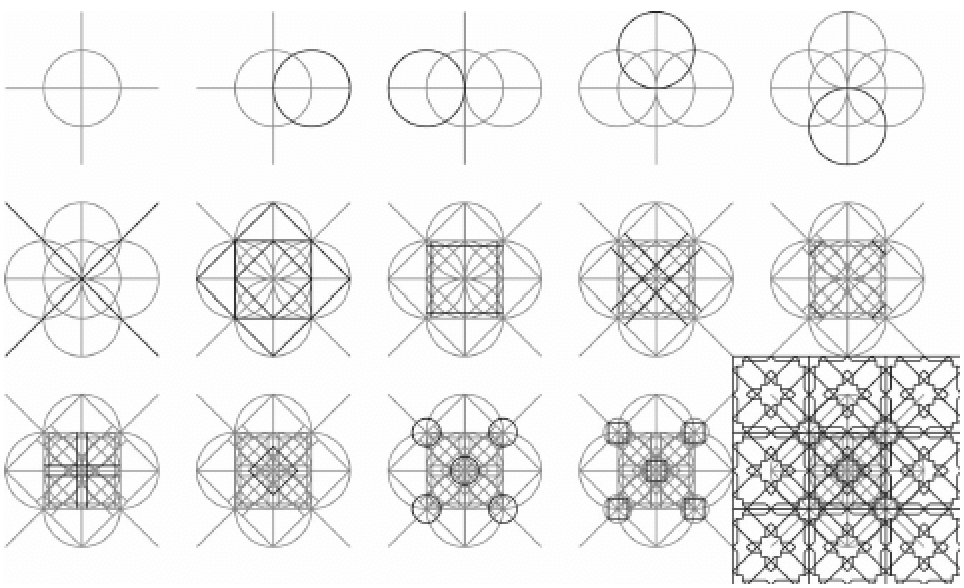


Figure3.

Methodology Flow Chart

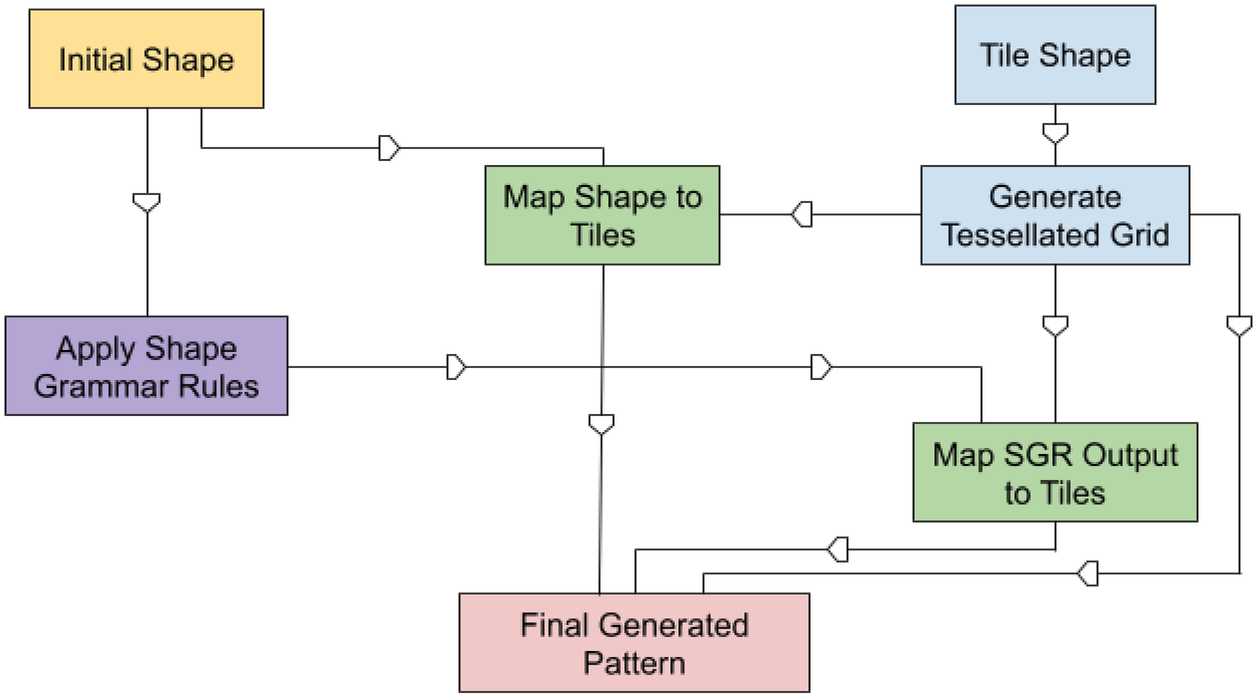


Figure 4.

Z. Sayed's shape grammar rules diagram (Z. Sayed, 2017)

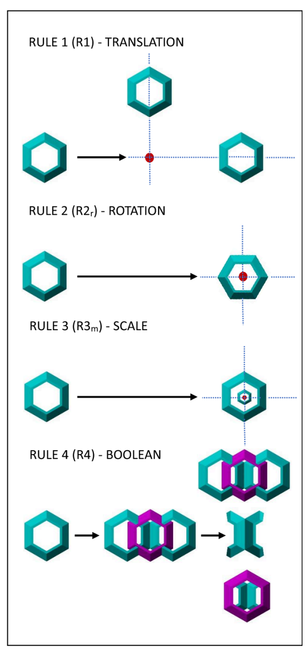


Figure 5.

Z. Sayed's applied shape grammar rules diagram (Z. Sayed, 2017)

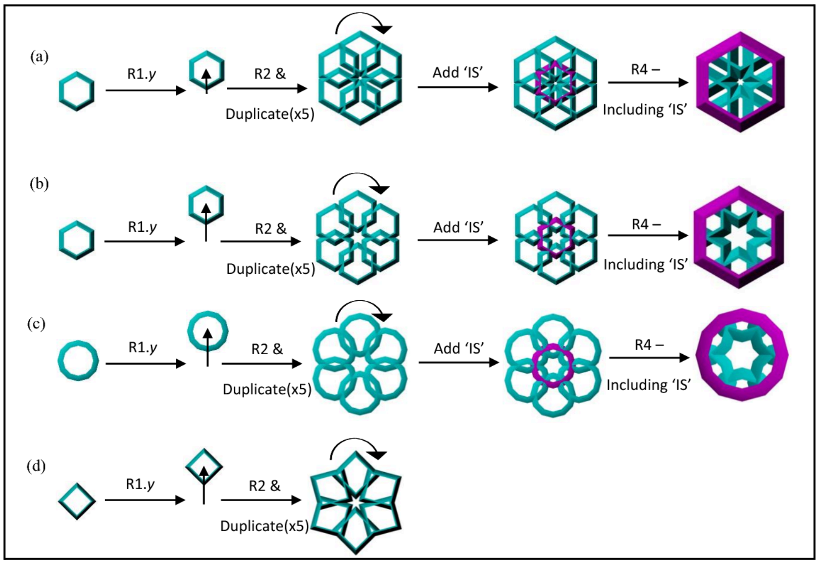


Figure 6.

User Interface, Pattern

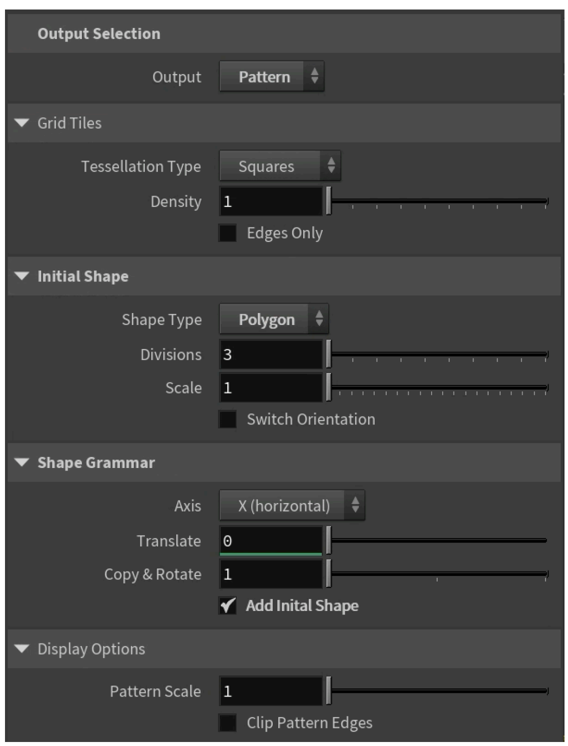


Figure 7.

User Interface, Motif

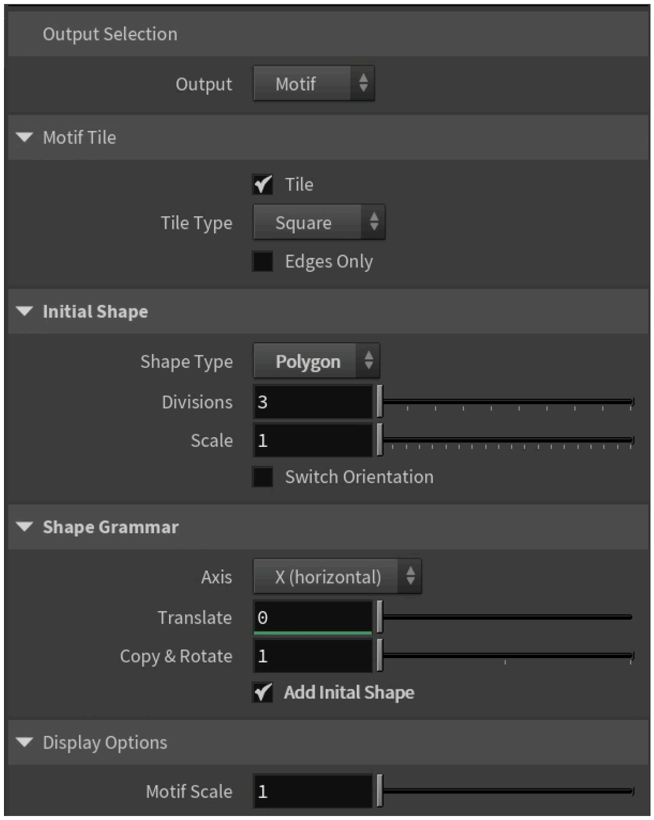


Figure 8.

Tool Network

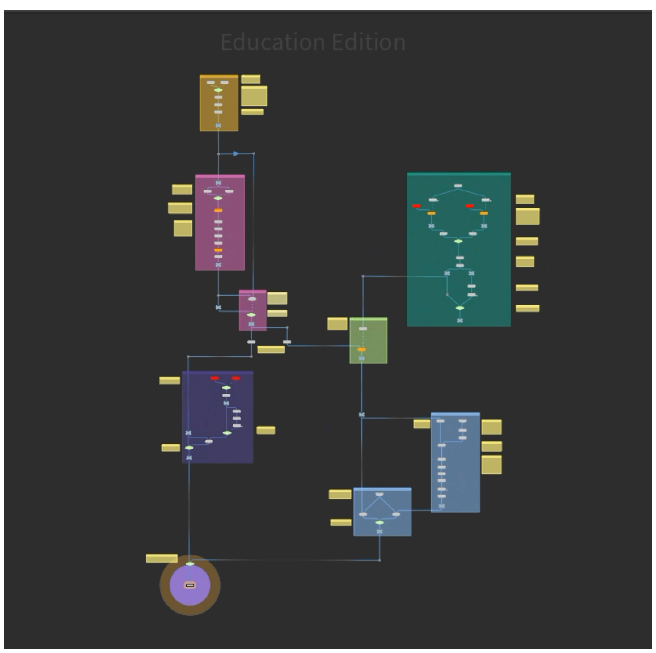


Figure 9.

Network- Initial shape generator

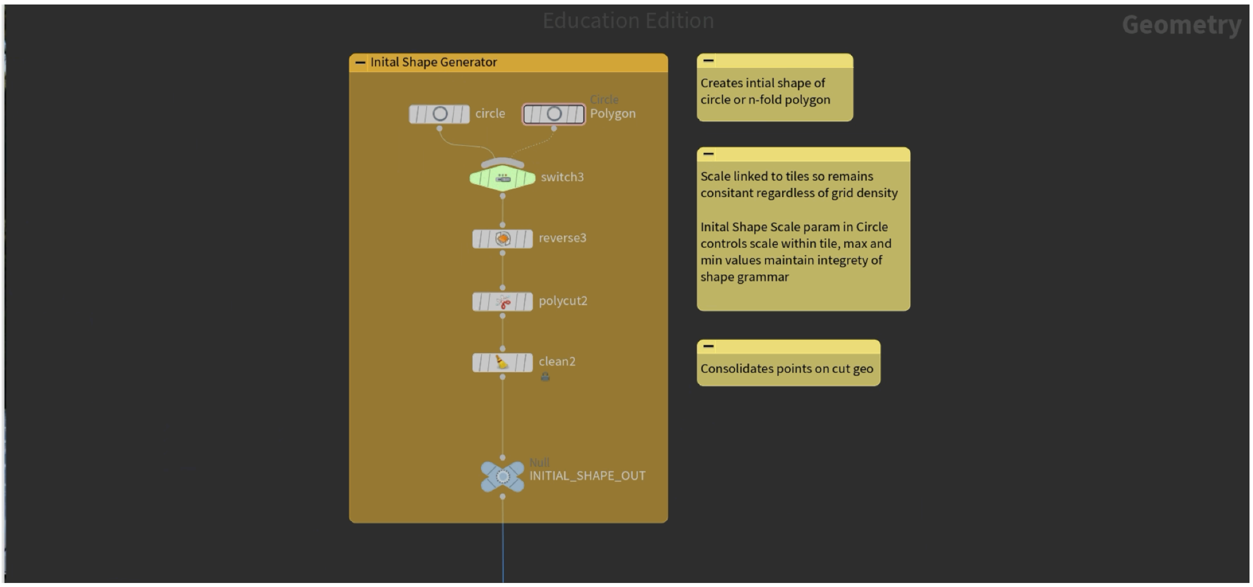


Figure 10.

Network - Tesselated Tiles Grid Setup

Figure 11.

Network - Shape grammar rules

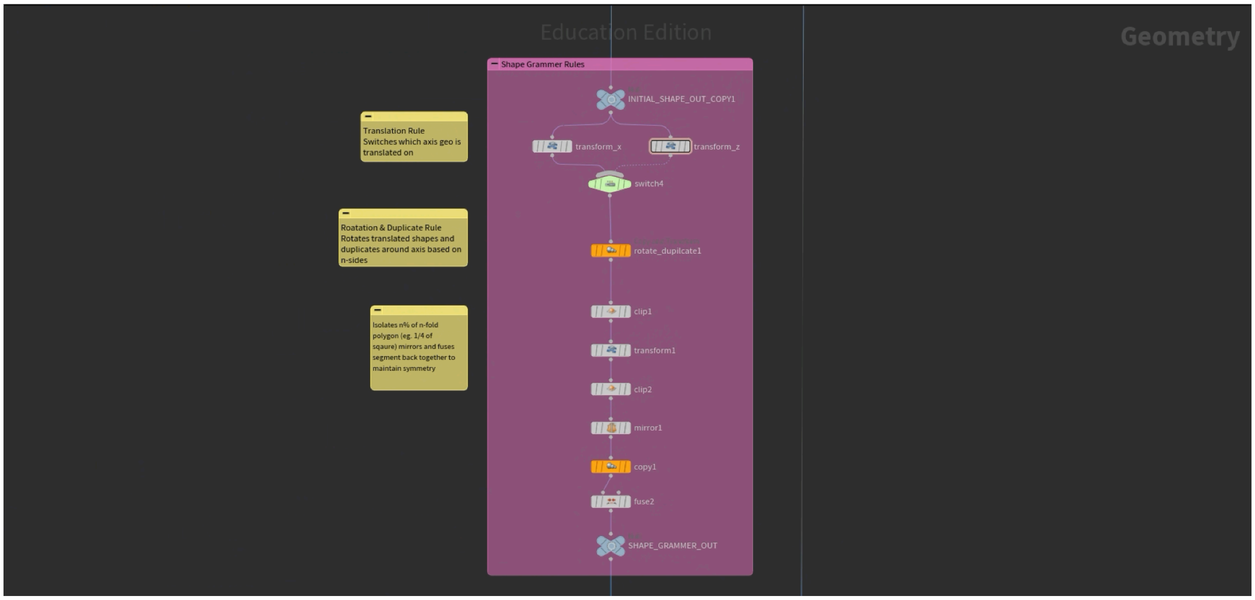


Figure 12.

Network - Add initial shape rule

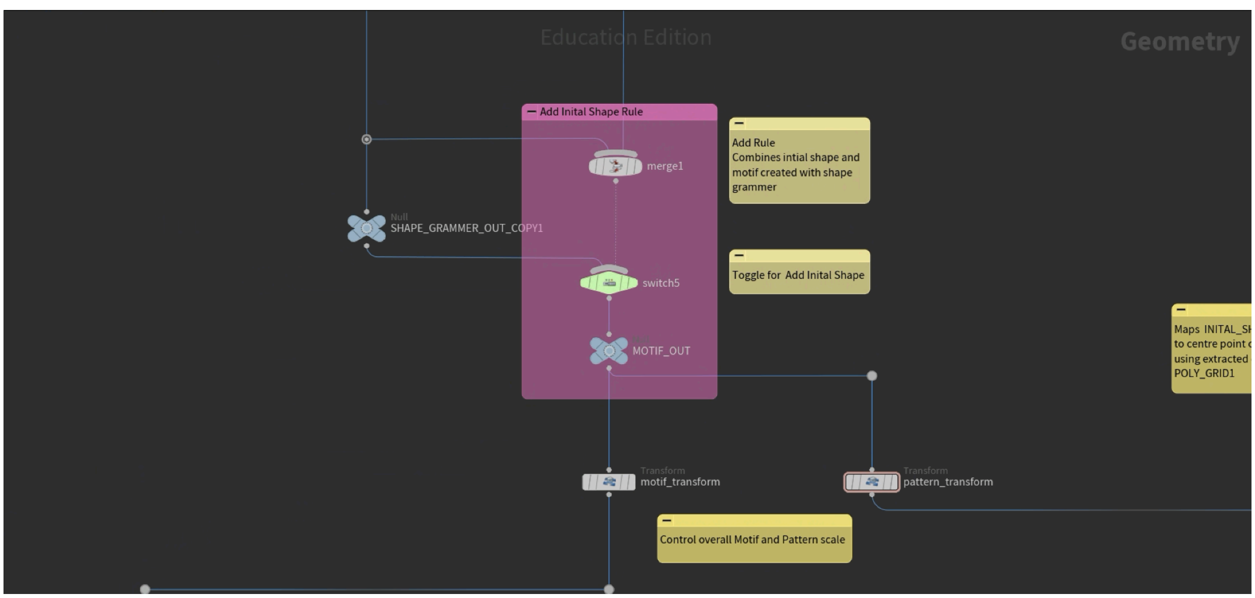


Figure 13.

Network - Map initial shape to tiles

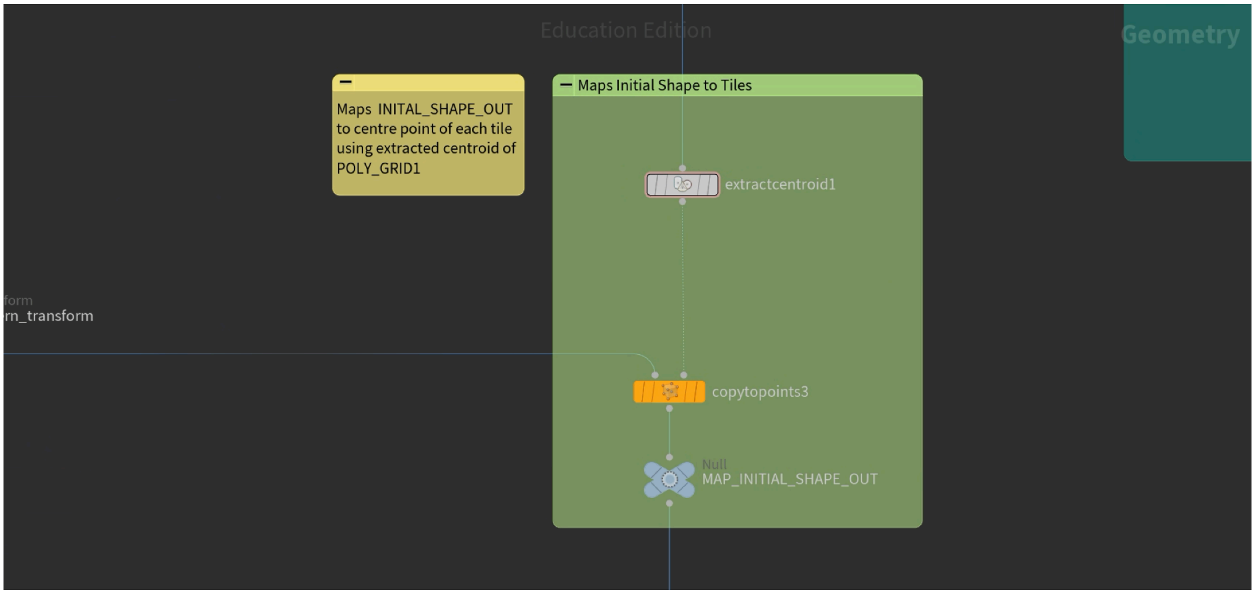


Figure 14.

Network - Tile selection for motif

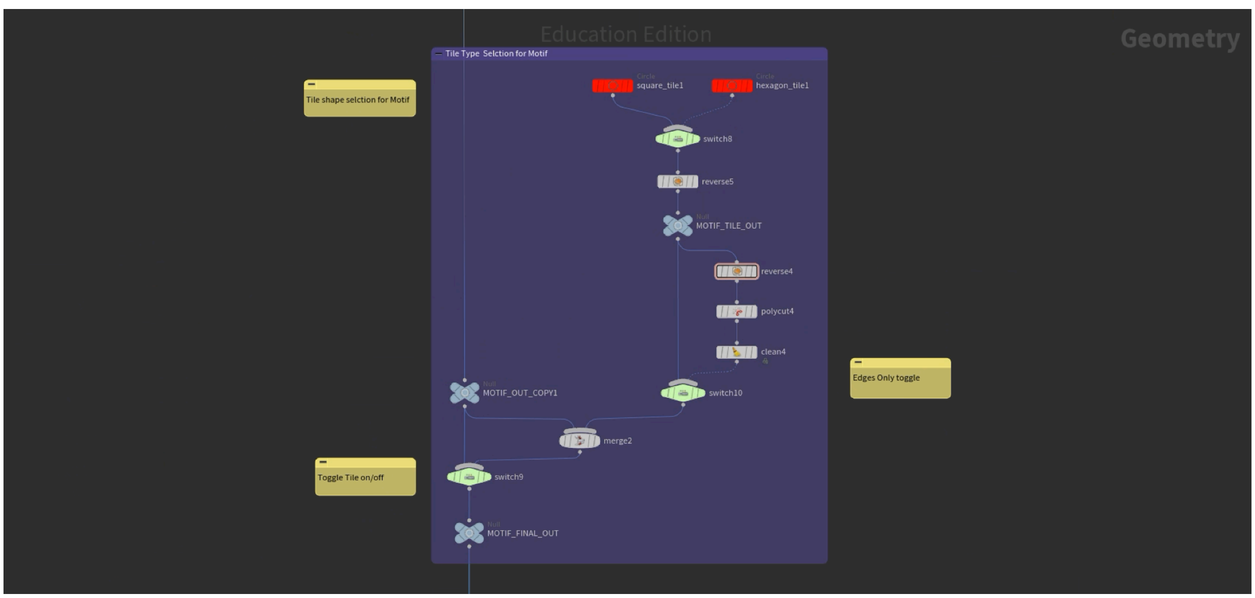


Figure 15.

Network - Clip pattern setup & toggle

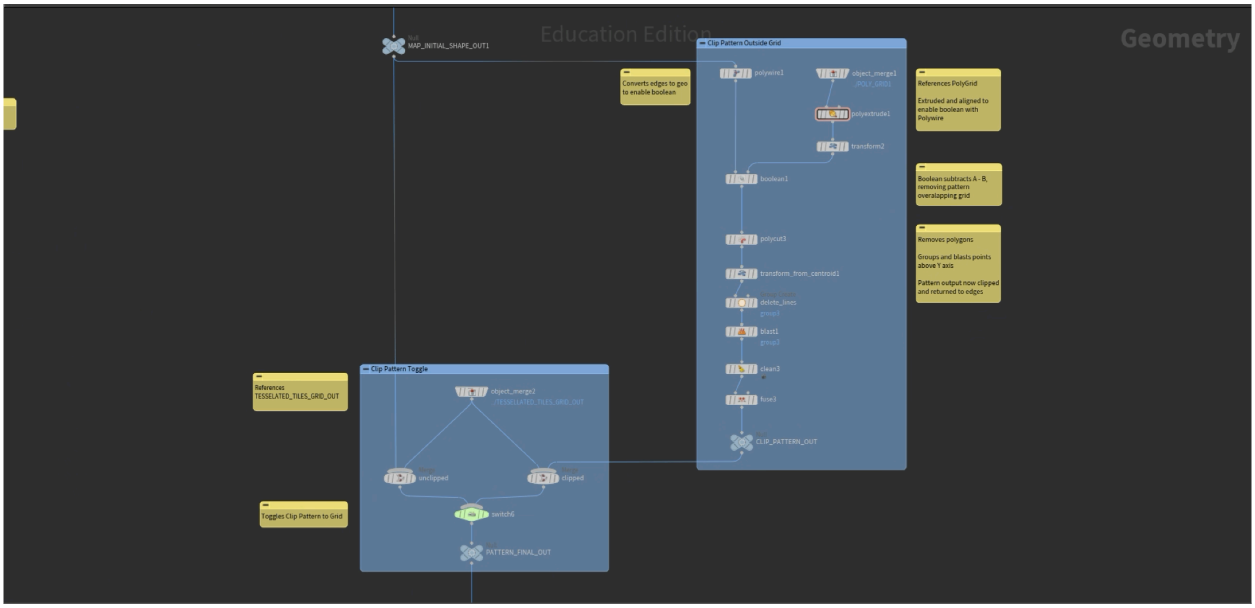


Figure 16.

Network - Motif/Pattern Output toggle

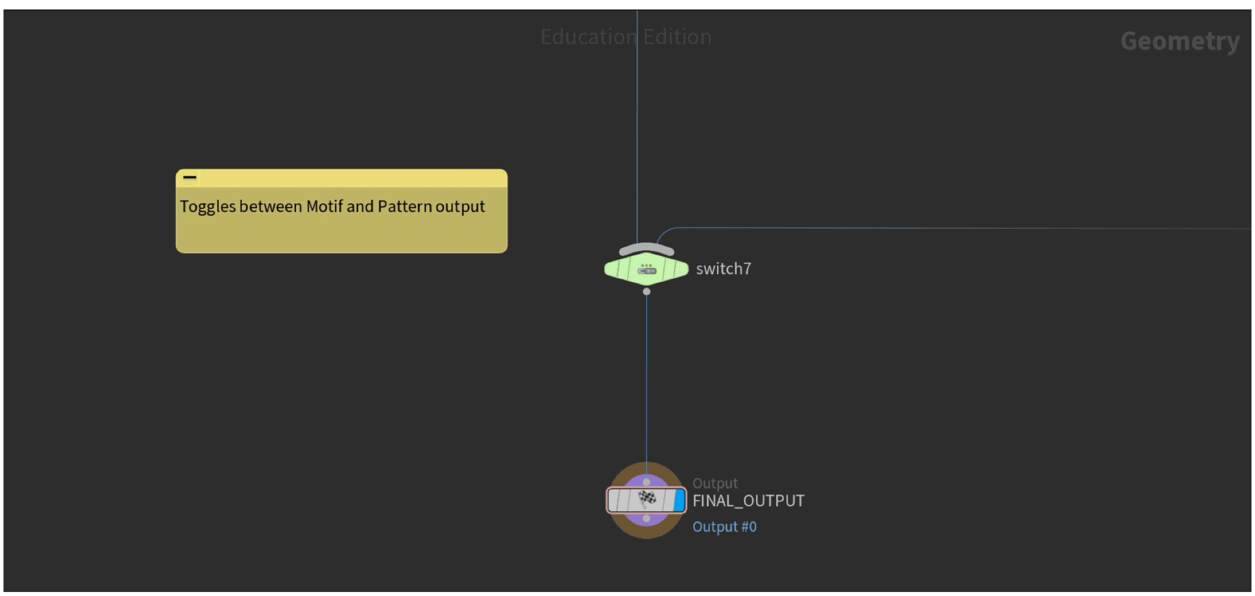
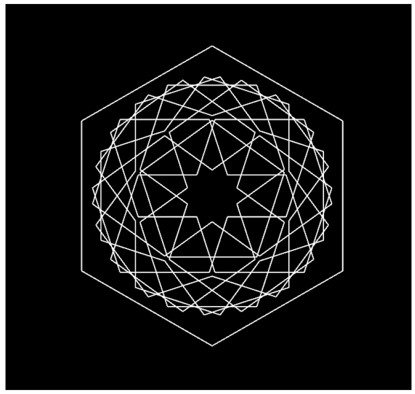
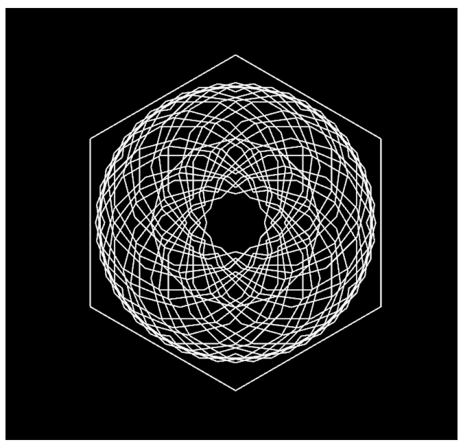


Figure 17.



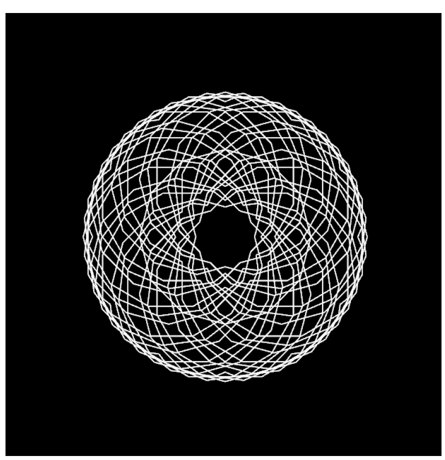
Motif A

Figure 18.



Motif B1

Figure 19.



Motif B2

Figure 20.

Pattern A

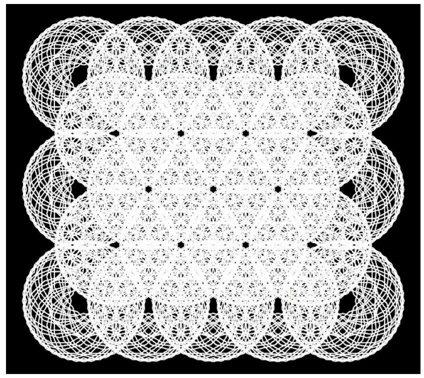


Figure 21.

Pattern B1

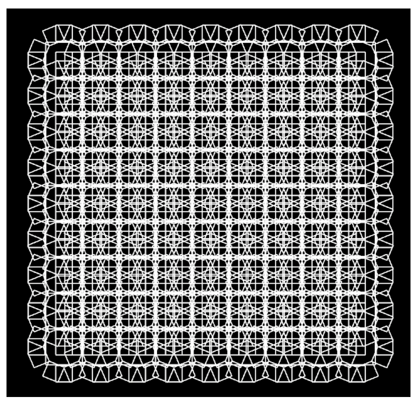


Figure 22.

Pattern B2

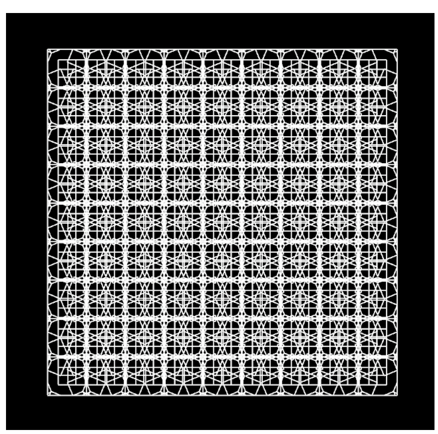
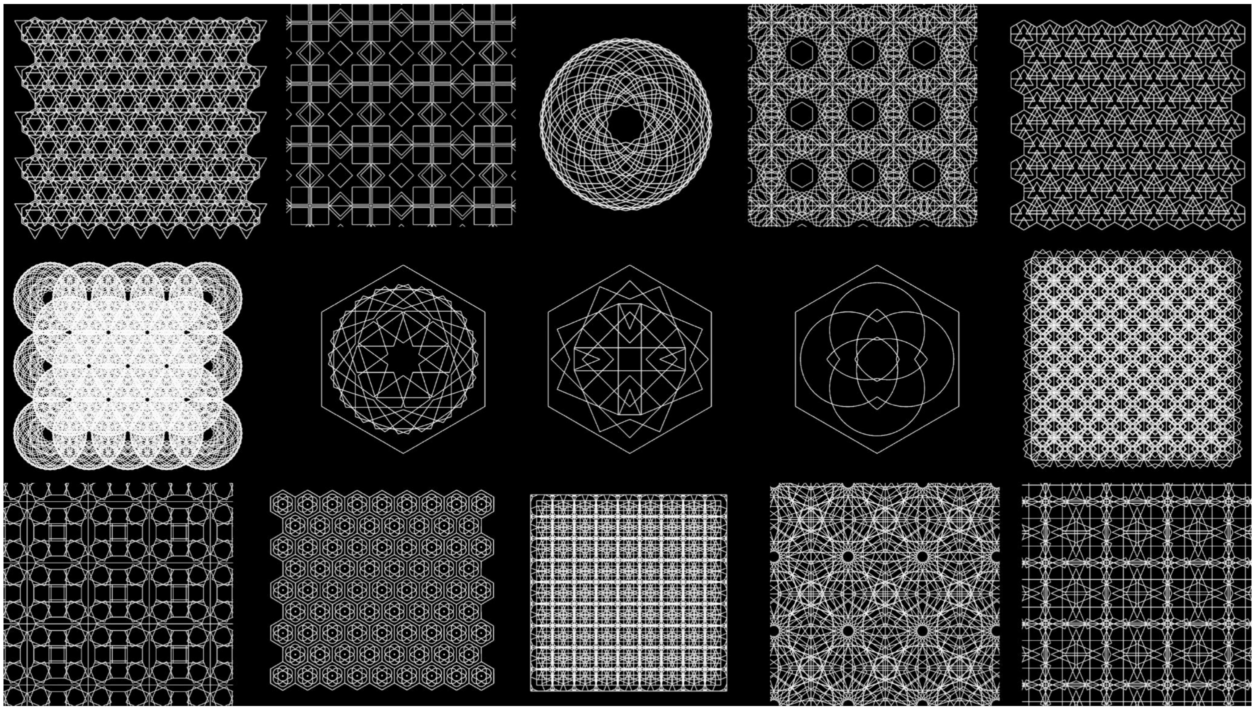


Figure 22.

Generated patterns and motifs



3. Introduction

3.1. Islamic Geometric Patterns Background

Islamic patterns are formed from regular geometric shapes (e.g. circles, squares and triangles) and transformed into intricate, tileable designs. These designs are constructed so that they can repeat infinitely, featuring interlacing lines and shapes to create varying levels of complexity. Islamic art avoids depictions of the divine and human figures and is characterised by calligraphic, geometric and floral patterns. The symmetry and repetition is spiritually significant as it represents the limitlessness, order and beauty of divine design. They can be seen in all forms of design: on tiles, mosaics, architecture, pottery, textiles, furniture and religious manuscripts.

Traditionally, these complex and time intensive designs would be made by hand utilising a ruler and pair of compasses. Today artists have access to advanced software such as Photoshop, Substance Designer, Maya, Houdini ect. enabling them to create work faster and more efficiently. However, these programs can be difficult to learn and not always artist friendly as they may require users to already be familiar with certain technical concepts or coding languages.

3.2. Aims and Objectives

This research aims to develop a proof of concept tool in the form of a Houdini digital asset to aid in artists creation of islamic geometric patterns. Houdini has been chosen for its powerful, procedural node based workflow, which allows the user to visualise the final result throughout development, and common use in industry. The focus will be on generating tileable islamic geometric patterns procedurally by analysing traditional methods of pattern construction, pattern generators, the mathematics of tiling and existing research. This tool should enable artists with no prior experience of Houdini software to focus on the creative process of design and iteration by utilising the adjustable parameters on the user interface. The UI should be intuitive and provide enough control over the final resulting pattern without breaking, producing void results.

I will build upon the work of Sayed, *(Z. Sayed, 2017)* using the outlined set of Shape Grammar Rules initialised by Stiny and Gips, *(G. Stiny, J. Gips, 1971)* to form the basis of the method of pattern construction. The scope of this project will be limited to generating 2D patterns on square or hexagonal tiles, varying levels of detail and creating outputs which the user could use to generate simple texture maps. In addition, the tool may aid in learning how islamic geometric patterns are traditionally constructed as one can isolate different stages in the network and the layout of the user interface is deliberately designed to reflect this.

4. Related Works

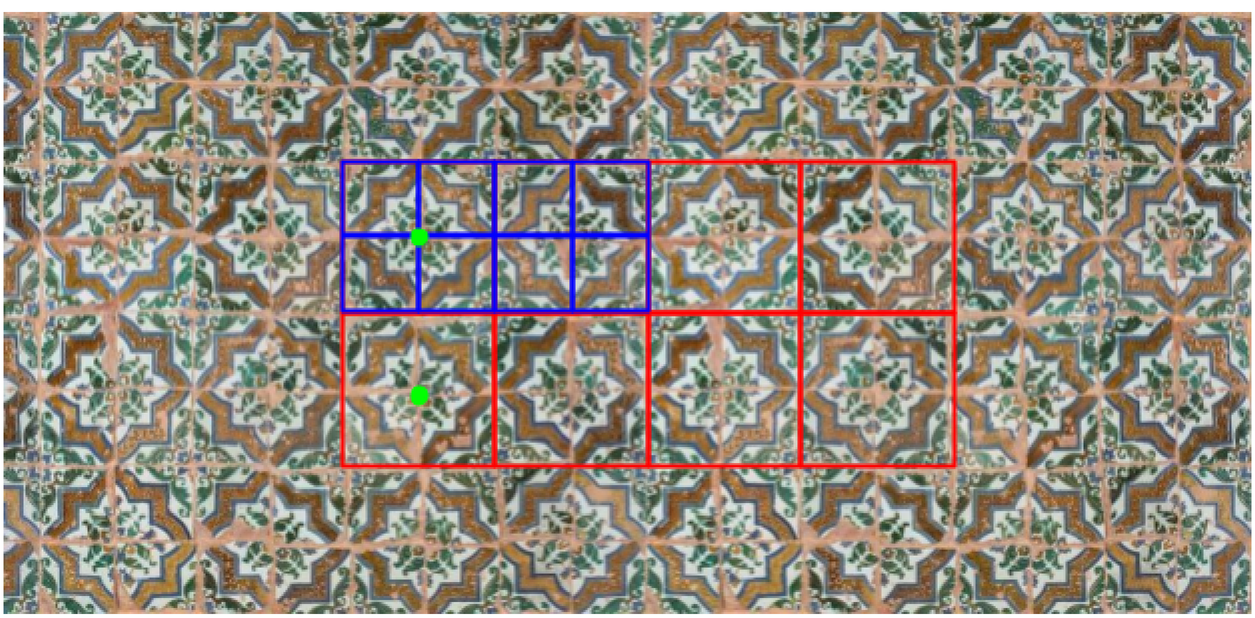
4.1. The Mathematics of Tiling

Tessellation, also known as tilling, is the covering of a surface with one or more geometric shapes referred to as tiles. Regular tessellation is defined by three sets of rules:

1. The tiles must be regular polygons
2. The tiles must be congruent (identical in size and shape)
3. The tiles must not have any gaps between each other or overlap

The most common shapes to be tiled are triangles, squares and hexagons. There are non-regular shapes that exist which can tessellate as well as semi-regular tilings (consisting of multiple shapes) but this research will focus on squares and hexagons due to their common use in islamic geometric patterns.

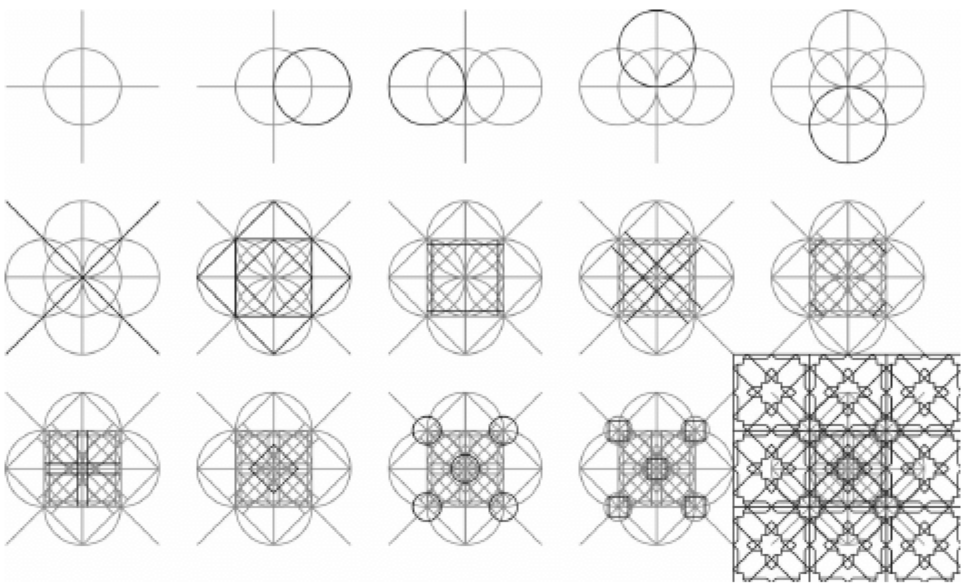
Traditionally a tile may be painted, carved, inlayed or created as a mosaic. The design may appear to overlap across tiles but each tile must be identical in order for the pattern to repeat infinitely.



Here a pattern was created using square tiles, rotated 90° to form a 8 pointed star. (Figure 1) It would also be possible to create this same design using a larger tile, as shown outlined in red, where the star motif is formed in the centre of the tile.

4.2. Analysing Islamic Geometric Patterns With Shape Grammar Rules

There is a wealth of existing literature on the mathematical analysis and implementation of islamic geometric motifs and patterns into computer graphics. These discuss the different methods of construction and varying levels of complexity which could be applied in their creation. Below is an example of the traditional ruler and compass construction method. (Figure 2)



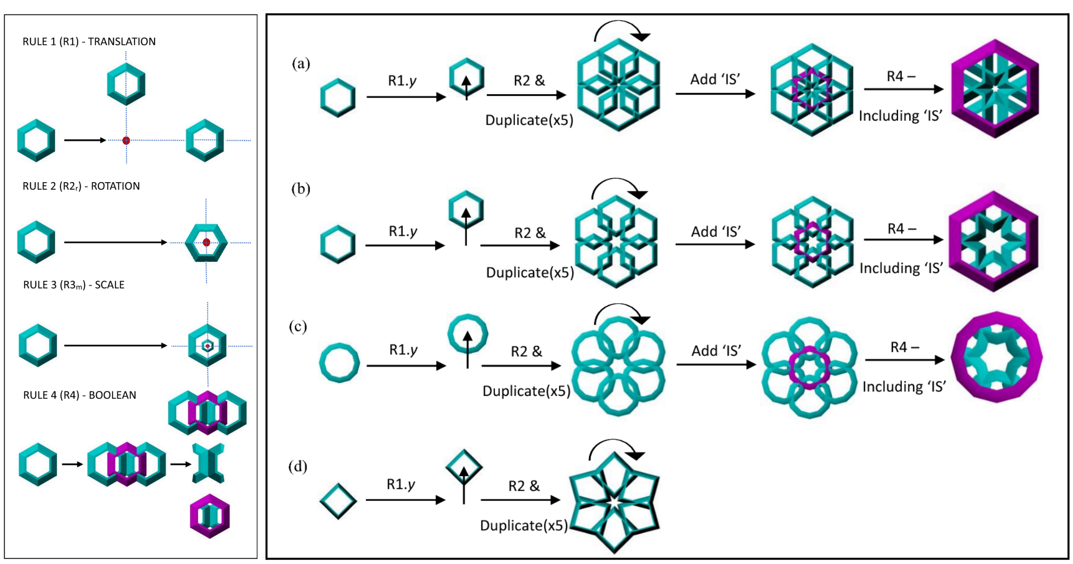
Jay Bonner is an architectural ornamentalist who is dedicated to revitalization of Islamic geometric design through the teaching of traditional methodological practices. In his book he discusses in detail the different classifications of construction via methodology; polygonal technique, point joining technique, the grid method and compass work.
(Bonner, J. and Kaplan, C.S., 2017) His work is aimed at designers rather than programmers or software developers but his explanations helped inform other researchers' algorithmic approaches to creating IGPs. It is important to consider both the mathematical aspects of tessellation and patterns as well as the artistic side. IGPs

show the harmony between maths, science and art. The symbolism is equally important as the mathematics behind their construction.

The circle is symbolic, signifying one God, and is the central shape from which all patterns are constructed. Three fundamental shapes are derived from the circle, which are also the three regular tileable polygons; the triangle, square and hexagon. Interpretations vary slightly but generally these symbolise: harmony, materiality and Heaven. (Z. Sayed, 2017)

A successful visual method of creating IGPs was developed by Sayed (Z. Sayed, 2017), adapting the shape grammar work of Stiny and Gips *(G. Stiny, J. Gips, 1971)*, applying their methodology to the different steps of the patterns mathematical construction, whilst maintaining the aesthetic and philosophy behind the designs. By analysing traditional construction methods Sayed developed the following visual shape grammar rules which can be applied to the initial shape of the design (regular n-fold polygon) in order to create an IGP:

- Translation
- Rotation
- Scale
- Boolean
- Add
- Duplicate

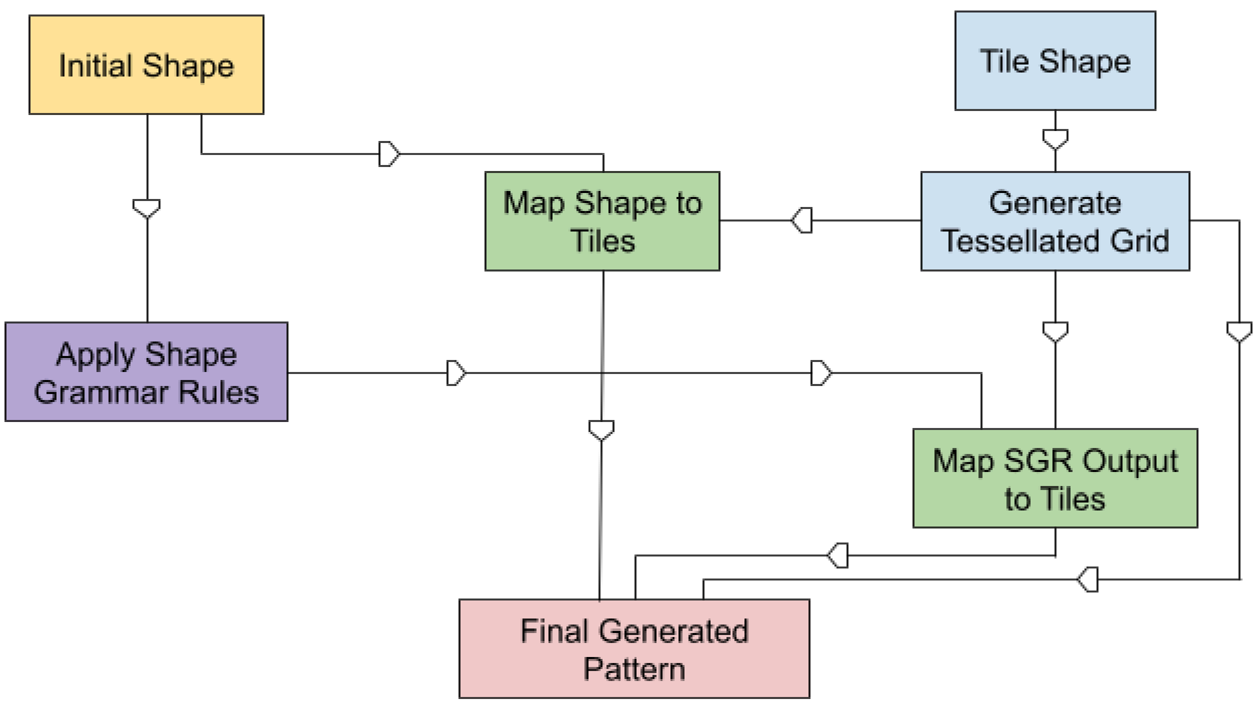


The diagrams above (fig. 4, fig. 5) show Sayed's shape grammar rules in practice, generating a variety of motifs. By parameterizing these rules, applying them multiple times and/or in a different order allows for variation in complexity and design.

5. Methodology

5.1. Defining Approach

Based on existing research I chose to adapt the shape grammar method devised by Sayed. This approach allows each stage of pattern creation to be visualised individually which works perfectly with Houdini's procedural node based approach. I developed a flowchart (Figure 3), shown below, outlining each function I had to create in my network. Once implemented into Houdini I would use these sections to build the user interface and select relevant parameters for the user to control final output.



5.2. Controllable Parameters

The user should be able to control the effect each shape grammar rule has on the design as well as any additional parameters that might affect the final result of the pattern. I organised each section of the UI by the order in which the pattern would traditionally be created (fig 3.); starting with the grid, the initial shape, shape grammar rules then any additional controls that didn't specifically fall within those parts of the network. The key controllable parameters were:

- The tiling shape (square or hexagon)
- Tile density (number of tiles in grid)
- The initial shape (circle or n-sided polygon)
- The initial shape scale
- Shape grammar rules (translation, scale, rotate & duplicate)

These were not the only parameters on the UI but formed the core controls available to the user after certain aspects were proceduralized. Additional controls were added later in development.

6. Implementation

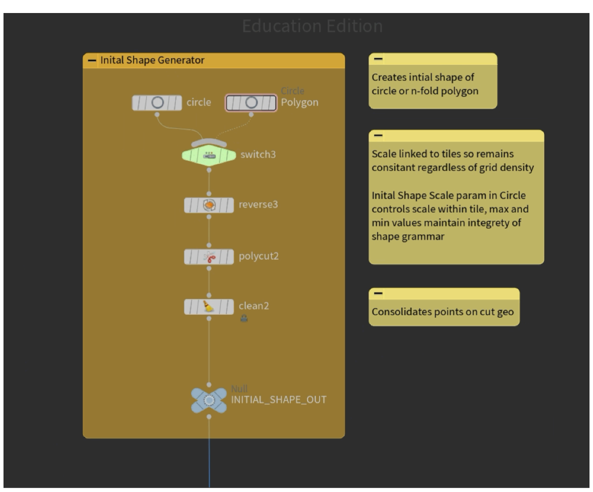
6.1. Creating Tessellated Grid

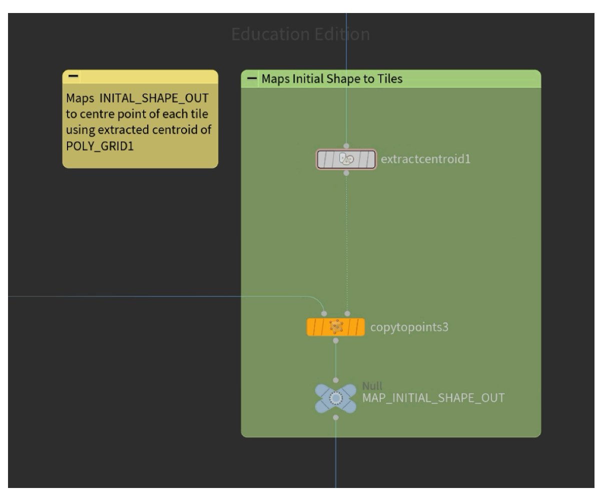
To begin I started by making a simple grid setup, as seen below, (Figure 10) to create and instance the geometry tiles; squares and hexagons. The setup had to allow for switching between the two tile types and tessellate seamlessly. This part of the network went through a lot of optimisation. Tiles are mapped onto points (grid or tetrahedral) then fused in order to remove any gaps or overlap. The hexagonal tiles required some extra work as the edge lengths were not equal after the fuse. An Edge Equalise node was applied after the Fuse which resolved the problem.

Secondly, I found that it was easier to visualise the patterns when the tiles displayed only their edges. This is able to be toggled on or off depending as it gives more flexibility to the user in how they wish to use the tool.

6.2. Mapping Initial Shape

In order to map the initial shape onto the tiles, I used a similar setup to grid. Taking the centroid of the geometry grid output and copying the initial shape output to points successfully mapped the shape onto every tile. These two parts of the network can be seen below. (Figure 9, 13)

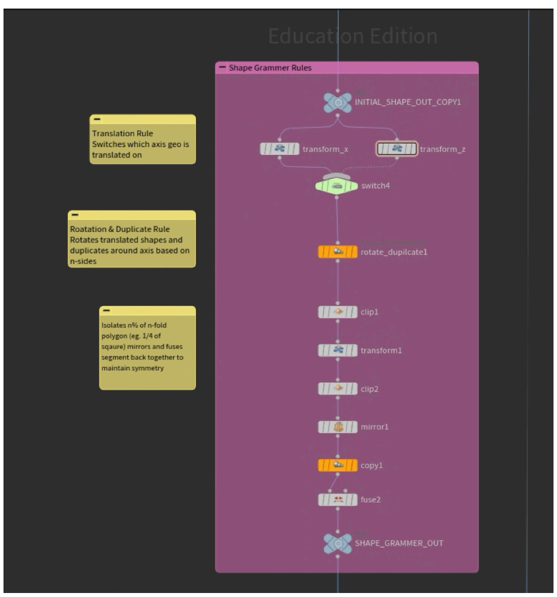


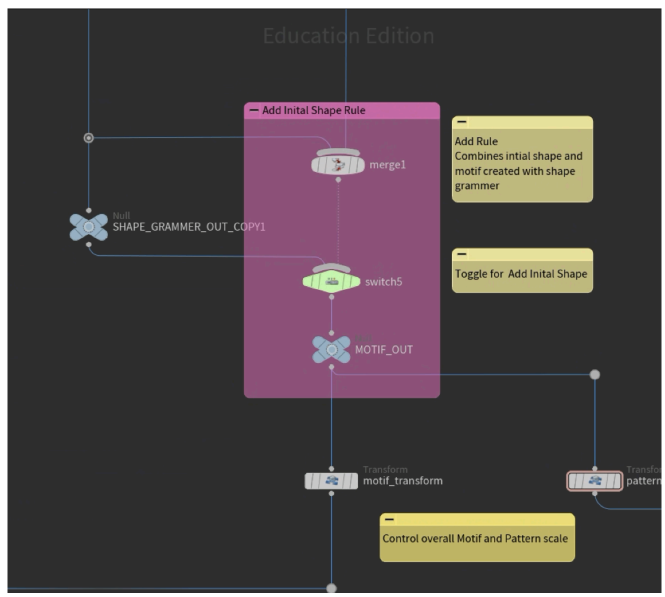


The initial shape's scale is linked to the grid via the Square tile so that when the density of the tiles changes the initial remains the same scale within each tile. An additional parameter was added to control the scale of the initial shape independently of the tile density. These parameters were added to the user interface under sections; Grid Tiles and Initial Shape.

6.3. Proceduralizing Shape Grammar Rules

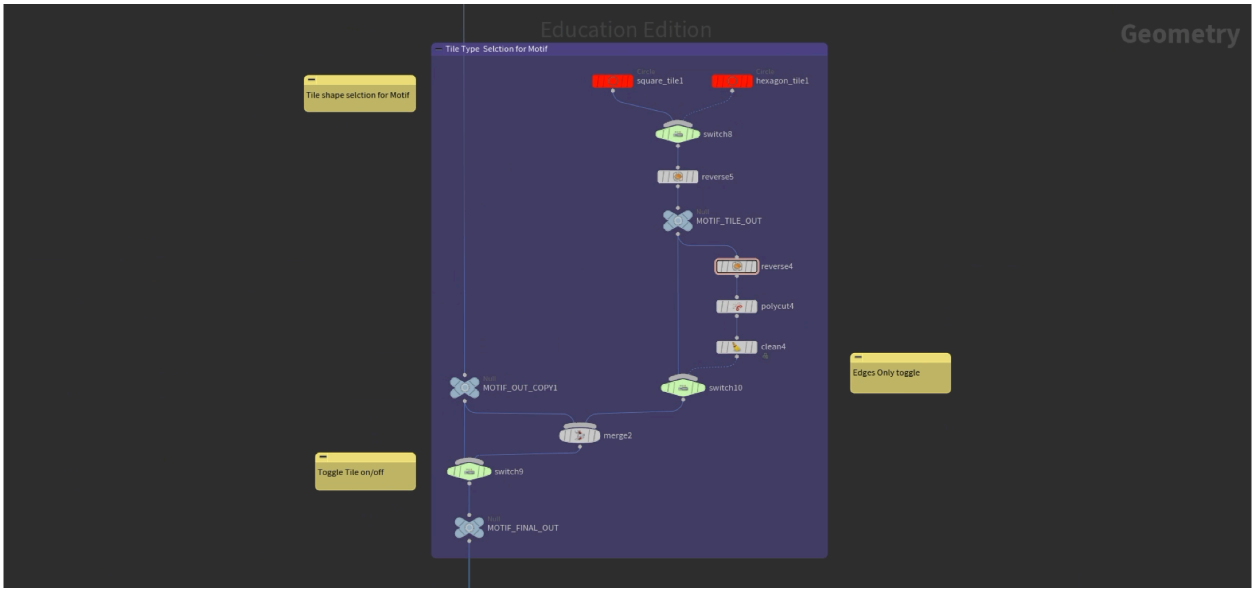
Implementing the shape grammar rules proved to be more challenging. Initially I wanted the user to be able to control all rules in the grammar: translation, scale, rotation, duplicate and add. However, this conflicted with the procedurality in a few ways; the amount of control I wanted to give the user was difficult to achieve without overcomplicating the network; this amount of control would also defeat the aim of the tool being artist friendly as they would now need to have prior knowledge of how the patterns are constructed in order to achieve the desired results.



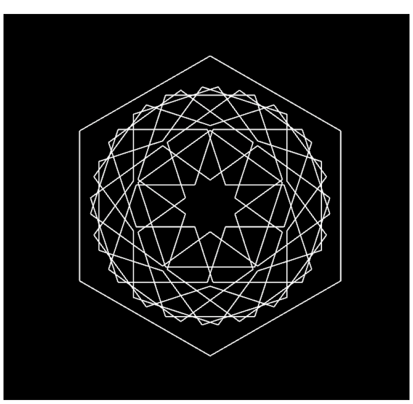
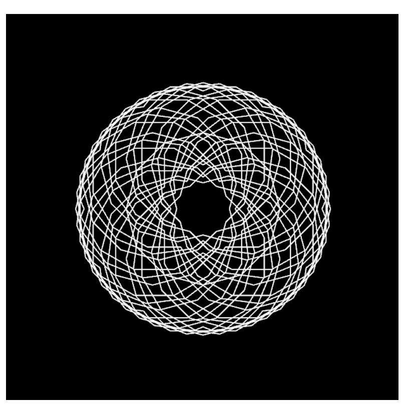
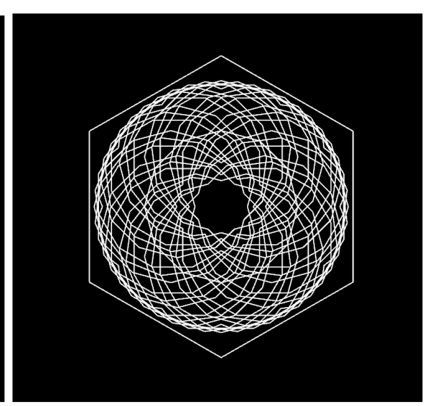


After experimenting with a few different setup I decided to embrace the procedurality and create a much simpler set of nodes. (Figures 11, 12) This effectively combined some of the shape grammar rules on the user end. Whilst some control was lost it wasn't significant enough to have an effect on the output patterns.

6.4. Troubleshooting Additional Features

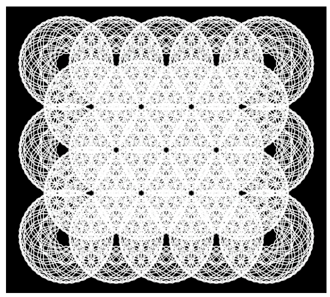


After building and debugging the basic tool, I decided to add some additional features to improve the variety of outputs that could be generated. Many of the previous works I had studied focused on creating different styles of Islamic geometrical patterns but did not have the option to generate both motifs and unique overlapping designs. By taking the output of the shape grammar rules before it is mapped onto the grid (Figure 14) the user is able to generate a motif rather than a tessellated pattern. In order to facilitate additional variety I recreated a version of the grid but with only a single tile. The motif output provides the same functionality as the pattern output, allowing toggle the tile to edges only. In addition, I added an extra switch that would allow the user to turn the tile on or off. (Figure 17, 18,19)

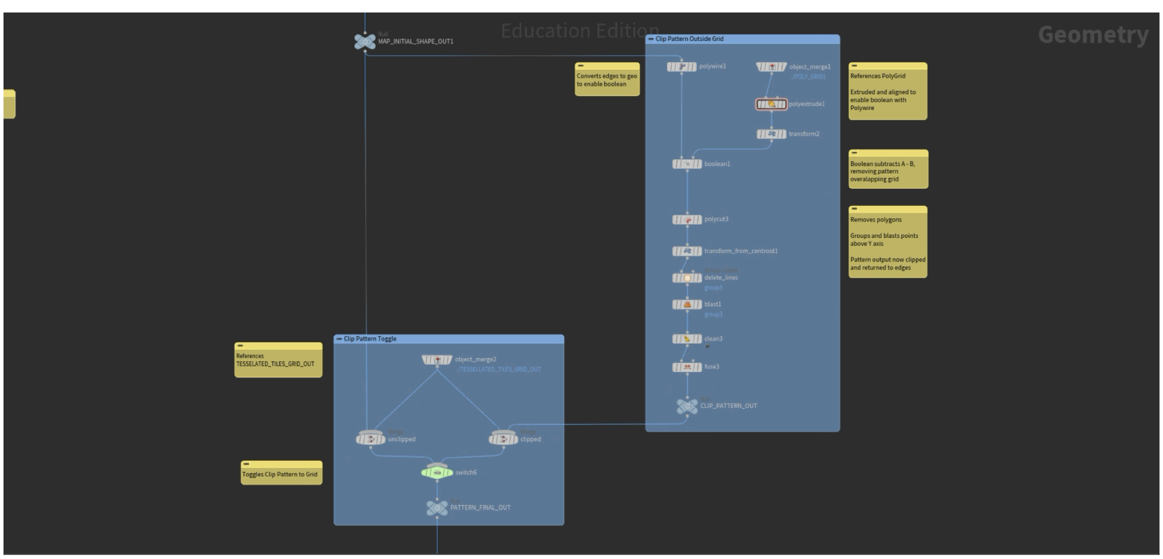




Flexibility was one of my key aims in creating the pattern generator, creating as many options without overwhelming the user. There are numerous features that could be integrated into this tool but I had to limit myself in order to achieve working proof of concept within the scope of the project.

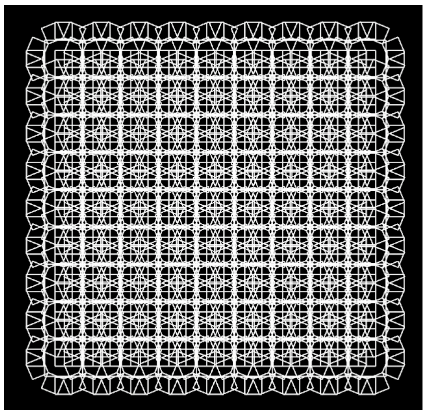
The final feature I added was the ability to trim/cut the pattern edges. Once the patterns began to overlap they created more intricate designs but also fell off the edges of the tiles. If a user wanted to create a texture map or use the tool to create effect geometry this would pose a problem. (Figure 20 below)

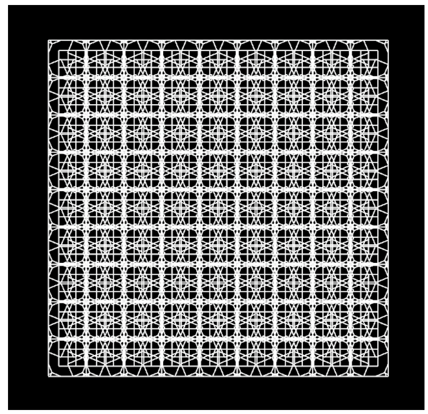


This was a lot more complex to implement than I had anticipated. (see figure 15 below) The mapped initial shape, tessellated tiles grid and the poly grid outputs all had to be plugged into this set up as the only successful method I found was to use the boolean node. This node requires geometry so the grid tiles had to be extruded and the vectors converted into polywire. In order to revert the pattern back into vectors I grouped and deleted all points of the polywire above the Y axis.



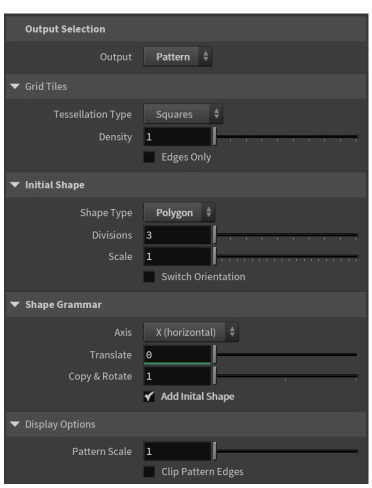
Whilst this did produce the desired result (see figures 21,22 below) it increased the loading times between user input. To solve this I used another switch node so the function could be toggled on or off. There is likely a more efficient method to achieve this result with more optimal loading times but I wasn't able to find a solution within the time of this project.

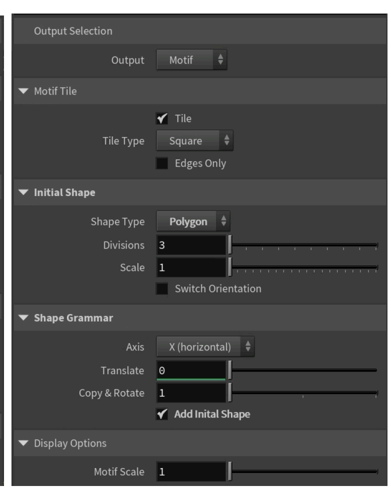




6.5. *User Interface*

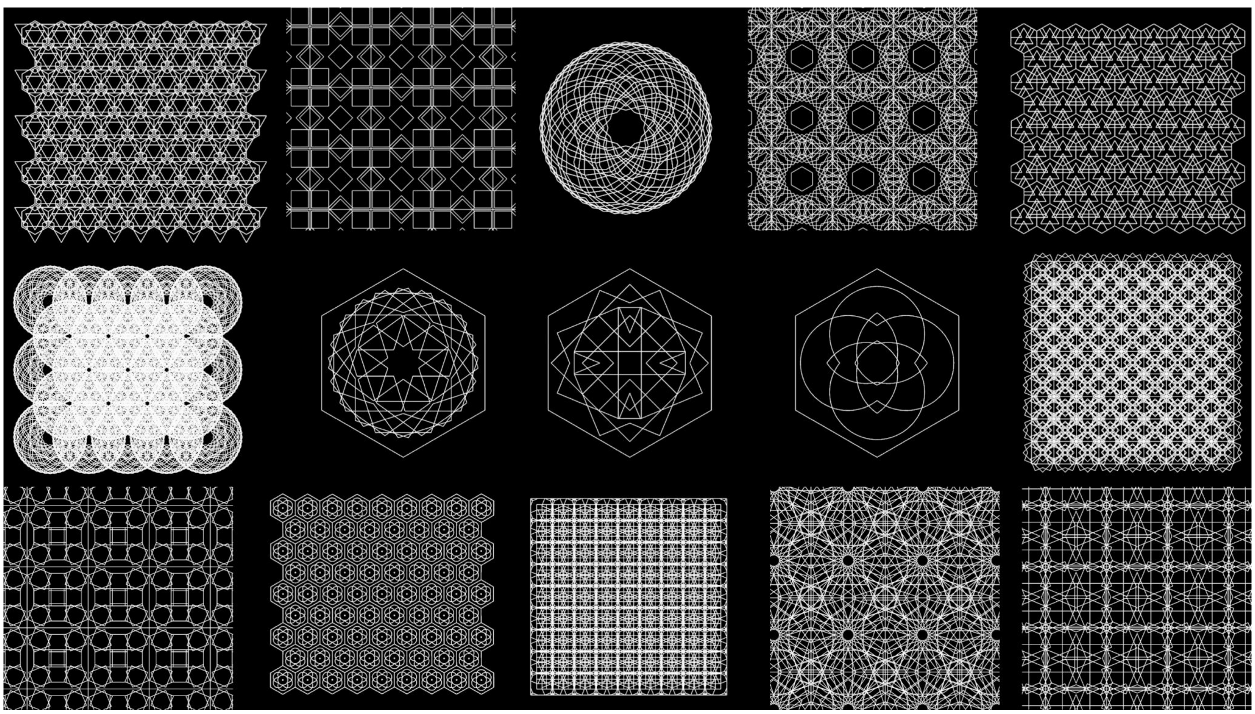
It was key that the user interface should be intuitive for all users so everything was named deliberately, striking a balance between technical and practical language. I believed it to be important that the user should be able to gain insight into the process of generating the motifs/patterns through the menu. Each part of the menu takes the user step by step through the adapted shape grammar method.





All parameter controls were chosen for optimal functionality and visual indicators; anything optional is a toggle, any key changeable elements are in a menu, everything that can be adjusted is a type of slider. When certain selections are made any irrelevant menu items or sections are hidden. Most parameters sliders are limited to prevent the user from inputting values that would produce void results or to prevent them from breaking the functionality of the generator. The menu could be improved by further limiting and adding additional parameters but most importantly every parameter acts as expected based on its label, description or type.

7. Results



The images above (Figure 23) show a small selection of the patterns and motifs generated with the tool. There was a fair amount of troubleshooting involved in regards to maintaining symmetry and the aesthetic of the patterns but the results are as expected. The tool proves that islamic geometric patterns can be produced procedurally whilst staying in keeping with islamic design aesthetics being artist friendly. The interface is easy to use and it follows the steps of traditional pattern creation as much as possible.

8. Conclusion and Future Works

8.1. Evaluation

In this paper I have presented my approach to develop an artist friendly tool to aid in the creation of islamic geometrical patterns taking advantage of the industry standard procedural software, Houdini. Considering this was my first time using Houdini I believe I was able to pick up the basics of the software and VEX for efficient application in this project. There was a fair amount of troubleshooting involved in regards to maintaining symmetry and the aesthetic integrity of the patterns which I felt was important to maintain. The majority of these issues were addressed and resolved. It is possible for the user to produce void or unexpected results through pushing certain parameters. I decided to leave this as by restricting the user to prevent this it restricted the variation in design that could be achieved. I felt it was more important to allow for iteration than try to prevent the user from breaking the patterns. However, with additional time it could have been possible to optimise certain aspects of the tool. Overall, this research successfully demonstrates how a procedural tool can be created for artists with little to no experience in the software to generate procedural islamic geometric patterns in Houdini.

8.2. Limitations

The project scope was limited to only allowing users to create square or hexagonal tiles and create vector based graphics. This was due to time constraints and my limited experience with houdini. However, this enabled me to focus on my key aim of creating an accessible tool for non-technical users. The "clip edges" parametre, required the vectors to be converted into a mesh in order for the boolean operation to function and then made back into a vector. When this is toggled on, the loading time increases significantly with denser and more complex patterns. However, this option is placed at the bottom of the user interface to encourage users to only apply when they have finalised their design. When it is toggled off any changes are instantaneous. There is no option to create 3D meshes with the tool. This was to maintain simplicity but could certainly be implemented into the tool given more time.

8.3. Future work

Future work could explore additional parameters to further the complexity and variation in pattern design, implement depth to generate 3D geometry, apply patterns to existing geometry to create embellishments or unique architectural features etc. There would also be ample opportunity to optimise certain aspects of the generator such as the boolean previously mentioned. In regards to the implementation I was able to achieve everything outlined in my aims and objectives. Whilst there are aspects that could be further developed and improved upon in future, the tool is successful as a proof of concept.

Bibliography

Abas, S.J. and Salman, A. (1992) 'Geometric and group-theoretic methods for computer graphic studies of Islamic symmetric patterns', *Computer Graphics Forum*, 11(1), pp. 43–53. doi:10.1111/1467-8659.1110043.

Andalucia 360 Travel Contributors (2023) 'Textures, colour and symbolism. The tiles of the Alhambra', *Andalucia 360 Travel*, 14 May. Available at: https://www.andalucia360travel.com/en/descubrir/tiles-of-the-alhambra/#ROUND_MAP (Accessed: November 2023).

Ahmed, M.O.M.M. (2022) 'Analytical Study of Muqarnas Formations in Islamic Architecture According to Digital Simulation', *Journal of Heritage and Design*, 2(7), pp. 315–342. doi:10.21608/jsos.2021.91995.1058.

Bellos, A. (2015) *Muslim rule and Compass: The magic of islamic geometric design*, *The Guardian*. Available at: https://www.theguardian.com/science/alexs-adventures-in-numberland/2015/feb/10/muslim-rule-and-compass-the-magic-of-islamic-geometric-design (Accessed: 17 January 2024).

Bonner, J. and Kaplan, C.S. (2017) *Islamic geometric patterns: Their historical development and traditional methods of construction*. New York, New York: Springer.

Caggese, M. (2020) *Material study in substance designer: Moroccan tiles*, *80lv*. Available at: https://80.lv/articles/material-study-in-substance-designer-moroccan-tiles/ (Accessed: 17 January 2024).

Claus, J. (2023) *Houdini tutorial: Creating escher inspired Tiling tesselations*, *YouTube*. Available at: https://youtu.be/2vfQ_df9PMY?si=QBJf-CoysyjDrFnQ (Accessed: 12 February 2024).

G. Stiny and J. Gips, "Shape Grammars and the Generative Specification of Painting and Sculpture," Int. Fed. Inf. Process. Congr., vol. 2, no. 3, pp. 125–135, 1971.

Jowers, I. *et al.* (2010) 'A study of emergence in the generation of Islamic geometric patterns', *CAADRIA proceedings*[Preprint]. doi:10.52842/conf.caadria.2010.039.

Lu, P.J. and Steinhardt, P.J. (2007) 'Decagonal and quasi-crystalline tilings in medieval Islamic architecture', *Science*, 315(5815), pp. 1106–1110. doi:10.1126/science.1135491.

Pérez-Gómez, R. (1987) 'The four regular mosaics missing in the Alhambra', *Computers & Mathematics with Applications*, 14(2), pp. 133–137. doi:10.1016/0898-1221(87)90143-x.

Rossetto, Q. (2022) *[eng] houdini tutorial - animated kaleidoscopic pattern*, *YouTube*. Available at: https://youtu.be/5H-XF-y3gmQ?si=685qlJy0DGfkHoyG (Accessed: 06 February 2024).

Sayed, Z. (2017) *3D mapping of Islamic geometric motifs*. thesis. University of Bradford.

Sayed, Z. *et al.* (2015) 'Parameterized shape grammar for generating n-fold Islamic geometric motifs', *2015 International Conference on Cyberworlds (CW)* [Preprint]. doi:10.1109/cw.2015.54.

Wikipedia Contributors (2019). *Islamic art*. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Islamic_art.