The University of Bradford Institutional Repository

<http://bradscholars.brad.ac.uk>

This work is made available online in accordance with publisher policies. Please refer to the repository record for this item and our Policy Document available from the repository home page for further information.

To see the final version of this work please visit the publisher's website. Access to the published online version may require a subscription.

Link to publisher's version: <http://dx.doi.org/10.1007/978-3-662-53090-0>

Citation: Sayed Z, Ugail H, Palmer I, Purdy J and Reeve C (2016) Auto-Parameterized Shape Grammar for Constructing Islamic Geometric Motif-Based Structures. In: Transactions on Computational Science XXVIII. Special Issue on Cyberworlds and Cybersecurity by Gavrilova ML, Tan CJK and Sourin A (Eds.): 146-162.

Copyright statement: © 2016 Springer-Verlag Heidelberg. Full-text reproduced in accordance with the publisher's self-archiving policy.

Auto-Parameterized Shape Grammar for Constructing Islamic Geometric Motif-Based Structures

Zahra Sayed^{*1}, Hassan Ugail¹, Ian Palmer², Jon Purdy³, and Carlton Reeve¹

¹ School of Media, Design and Technology, University of Bradford, Bradford, UK
`*z.sayed1@student.bradford.ac.uk`, `(h.ugail,c.reeve)@bradford.ac.uk`

² School of Creative Industries, Pearson College, London, UK
`ian.palmer@pearson.com`

³ Department of Computer Science, University of Hull, Hull, UK
`j.h.purdy@hull.ac.uk`

Abstract. The complex formation of Islamic Geometric Patterns (IGP) is one of the distinctive features in Islamic art and architecture. Many have attempted to reproduce these patterns in digital form, using various pattern generation techniques, in 2D. Shape grammars are an effective pattern generation method, providing good aesthetic results. In this paper we describe a novel approach in generating 3D IGP using the shape grammar method. The particular emphasis here is to generate the motifs (repeated units with the pattern) in 3D using parameterization. These can then be manipulated within the 3D space to construct architectural structures. In this work we have developed two distinctive Shape Grammars in 3D namely Parameterized Shape Grammar (PSG) and Auto-Parameterized Shape Grammar (APSG). Here the PSG generates the motifs and the APSG enables construction of the structures using the generated motifs. Both grammars are practically implemented as a 3D modelling tool within Autodesk Maya. The parameterization within each grammar is the key to generate both Islamic geometric motifs and Islamic geometric motif-based structures.

Keywords: parameterized shape grammars; pattern generation; shape modelling; architecture; 3D; Islamic art

1 Introduction

Geometric patterns are seen everywhere with their characteristic of regularity, both in nature and manmade designs. One art that majorly comprises geometric patterns is Islamic art, and these are known as Islamic Geometric Patterns (IGP). IGP are distinctive in design and are based on specific geometric shapes. They are a category in Islamic art functioning as a major element of adorning architectural surfaces (see Fig.1). Islamic art is distinctive for using a non-figurative approach. It is a way of showing the unity of God, replacing many figurative images with a single perception of God [1].

Fig. 1. Geometric pattern on the window in Qutb-Minar, Delhi, India

From the very beginning, in Muslim societies, geometry was seen to be a unifying structure, an art form of the unseen in Islamic art. Jowers et. al. describes how visual statements of the Islamic religion are created by the use of geometry in Islamic art [2]. For example, the famous 8-pointed star, which is known as the Khatemi Sulemani (Seal of Solomon), symbolizes the eight bearers of God's throne [3].

There are many different types of IGP. Some are simple, like the repetition of regular polygons, and some are more complex like the overlapping and interlocking of geometric shapes. One of the unique features of IGP is the star that is central to the pattern. IGP usually contain 6-, 8-, 10- or 12-pointed stars, although it is not uncommon to see odd number point stars like 7- or 9-pointed stars or even a large number of points such as a 20-pointed star. It is worth noting that, all stars are regular, with the same angle of degree and have at least one rotational symmetry as symmetry is an important characteristic in IGP [1].

As mentioned, symmetry plays a vital role in IGP and how they are formed. It unifies the patterns by creating a sense of balance and unity. Symmetry in IGP can be categorized into three different classes. They are [4]:

- *symmetry in a single direction (repetition of motifs)*
- *symmetry in two directions (tiling of motifs)*
- *radial symmetry (symmetry around a central point)*

Complex IGP that are constructed by the overlapping of shapes create emergence. Emergence is another key characteristic of IGP. It can be defined as shapes not present in the initial building blocks that are identified either during a pattern generation or when the final pattern is completed [2]. As the geometric shapes are overlapped or tiled in a particular ordered way, the shapes emerge automatically.

Self-similarity, usually seen in fractals, is where the shape is scaled proportionally exhibiting the same form. This is also a characteristic employed in IGP.

The scaling of the geometric shape or central star motif is what produces self-similarity in the derived pattern.

Our work presents a novel approach for generating IGP in 3D. It adopts the shape grammar method and uses parameterization. Previous works have utilised methods which generate IGP with flexible or non-constrained environments, either by using a polygonal network as a guide [5] or regenerating IGP as a set of modular tiles to build a specific type of IGP with no specific constraints [6]. We show how numerous original and novel motifs can be generated in 3D by a single regular polygon with the use of Euclidean transformations and parameters without any polygon grid or guide. We further show how utilising the 3D motifs we can build 3D architectural structures.

The remainder of this paper is structured as follows. Section 2 reviews the related work on construction methods, both classical and computerized, for generating IGP. Section 3 presents our parameterized framework for constructing 3D motifs and their related architectural structures in 3D. The results of this work are presented in Section 4. Finally we discuss the conclusions and possible future work arising from this work in Sections 5 and 6 respectively.

2 Related Work

The intricacy and complexity of IGP show how the medieval mathematicians and craftsmen were very skilled and learned people. The exact methods used to construct an IGP are still however not fully known. This is partly due to the secrecy of the skill being passed onto the student of the learned only [7].

The task for creating patterns digitally can be tedious if carried out manually. Many different methods have been implemented for producing patterns in computer graphics [1], [5]. As far as the creation of IGP are concerned, many approaches in computer graphics have been used in the past. These range from forming classifications of IGP [7] to creating algorithms for constructing IGP [1], [5], [9].

2.1 Construction Methods

Different techniques and methods are employed to construct IGP that have been developed from the evaluation of IGP [5]. Despite not knowing the actual construction methods, the intricacies of the process used are still researched today. In this subsection we describe three typical methods currently utilized to create IGP which are:

- *Strap Work method*
- *Modular Systems*
- *Polygons-in-Contact (PIC)*

Strap Work method: Until recently, research in constructing IGP was proposed by the theory of using the ‘Strap Work’ method. The Strap Work

approach uses two simple tools i.e. a straightedge (ruler) and a pair of compasses. Using this method, IGP are created out of shapes such as triangles, squares, circles and polygons that are formed by straight lines and arcs produced by the compass angle. By using these two tools one can create shapes which are then transformed into stars and overlapping lattices [10]. Although the use of a straightedge and a compass sounds easy, the process can however become cumbersome and errors such as angular distortions can accumulate during the process. Errors can also be introduced when repairing a damaged section [11].

Modular systems: A modular system is a pattern made up of smaller sets of modules or shapes [12]. Modular systems are known to be one of the famous traditional methods for creating IGP in the western part of the Islamic world [12]. They are usually placed to create a square or rectangular motif that creates a tessellation when repeated periodically, based on a square or rectangular grid, respectively. There are two main benefits in using this method. Firstly it is easier to produce a large number of motifs. Secondly the process is simpler because it uses a set of modules. This simplicity also has an aesthetic appeal to the viewer [12]. However the modularity also has a drawback: the system is made up of many small individual shapes. To create the pattern one would need to recreate the modules and have an algorithm to calculate the positions of where the shapes would be placed in order to reproduce the pattern, but again the system would need to generate a polygonal network initially for the algorithm to work.

Polygons-in-Contact (PIC): PIC is another method for constructing IGP. It was first initiated in the West by Hankin [5], [12] who discovered a polygonal grid under one of the original IGP on a monument in India [13]. This method uses a network of polygons that acts like a base to generate the IGP. The polygons are connected by their edges to form a polygonal network. Regular tiling and semi-regular tiling usually make up the polygonal networks. The IGP are then generated within the network base. Kaplan [5] derieved a computerized algorithm by integrating the PIC method to form automated 2D IGP. One advantage of this algorithm is the possibility for the user to change the contact angle that provides the possibility of creating a variety of IGP. It however, needs a polygonal base to initiate the generation of the motif.

In contrast with the methods described, our method generates Islamic geometric motifs without a polygonal grid or a set of modules. The motif is created with a single regular polygonal, with paramaterized shape grammar rules, making it easy for the user to generate motifs in a 3D environment.

2.2 Use of Shape Grammar

Shape grammars were first used by Stiny and Gips [14]. It is a method based on shape rules such as addition and subtraction of shapes, as well as Euclidean or other complex transformations [15]. This method is used to create both original and new designs. Stiny devised shape grammars as mathematical rules, describing the final design formation through an algorithm using shapes and rules, hence creating them in algebras of shapes [16]. A shape grammar (SG) is defined by the following 4-tuple, $SG = (V_T, V_M, R, I)$ [14] where:

- V_T is a set of terminal shapes
- V_M is a set of markers (non-terminal shapes)
- R is a set of shape rules, $u \rightarrow v$
- I is the Initial shape

where, V_T is a set of shapes which are non-changeable i.e. symbols [15]. V_M are spatial markers which are used for assembling the shapes in their positions. The shape rules, R , which are made up of u and v are in $(V_T \cup V_M)^+$ and $(V_T \cup V_M)^*$ respectively, where $+$ excludes an empty set and $*$ includes an empty set. The initial shape I is made up of both V_T and V_M .

The initial shape defines the final design but it is the rules that are the key to forming the end design [2]. Varying the rules on the same initial shape forms various different designs. The amount of times the rules are used varies the complexity of the final design which causes emergence in designs [16].

Shape grammars have been used in various areas such as building architectural monuments [17]-[19], historical ornaments [20], [21] and many works in pattern generation [16], [20], [22]. There are notable examples of the use of shape grammar for creating IGP [2], [22]. For example, Ulu and Sener constructed a grammar to produce IGP in 2D using a decagon as an initial shape [22]. The grammar however formed an outline of the pattern. The IGP design itself was generated using modular systems.

Shape grammars can also be parameterized by subdividing the rules into further rules [17]. This gives the advantage of allowing the designer to create patterns with simple transformation rules with the flexibility of parameters. For example taking the rule of translation from a point a to a point b , if the rule is parameterized one can apply modified rules to generate more designs than initially possible with a non-parameterized rule.

The following section describes the framework and methodology we have developed to generate both 3D n -fold Islamic geometric motifs and 3D structures, with the use of a single regular polygon.

3 Framework

Although shape grammar has been a method that has been applied to create IGP previously, the method has not been applied to create complete forms, or even a numerous amount of IGP. From all the pattern generation work described above, none of them create a 3D IGP which has generalized shape grammar rules. In this work the shape grammar is devised to create complete forms of n -fold Islamic geometric motifs in 3D and also has the advantage that they can generate novel forms using these rules. The generated motifs are then manipulated to create a tessellated surface in 3D space, in various forms.

The methodology adopted in this paper implements a parametric shape grammar approach, generating original and novel motifs with a single regular polygon and then manipulating them to create 3D architectural structures. The entire system is made up of three important steps (see Fig.2). They are:

- *Select Initial Shape (IS)*
- *Apply Parameterized Shape Grammar (PSG) rules to generate motif*
- *Apply Auto-Parameterized Shape Grammar (APSG) rules to construct structure*

Initial Shape (IS): The IS is the starting point for constructing Islamic geometric motifs. It is the shape that forms the motif. For example to construct a 6-pointed star the first step would be to start off with a regular polygon, e.g. a hexagon. In the methods described in the previous section, only one initial shape was adopted, allowing the grammar to produce patterns with a single shape. In our system, the initial shape can be changed, depending on the users interest. The initial shape is however constrained to a regular polygon of n -sides, as IGP are created with regular polygons.

Fig. 2. A tree diagram representing the framework of the system to construct Motif-Based Structures.

Parameterized Shape Grammar (PSG) rules: Following the example of the 6-pointed star the shape requires rules to transform it into a derived design or shape. To form the 6-pointed star, rules of translation and rotation need to be applied. These rules are in fact parameterized in our shape grammar to form the parametric shape grammar rules. The parameterization gives the advantage of generating numerous patterns with the single same initial shape. For example this can be achieved by varying its size (using the scale rule), or changing the parameter of the translation rule.

Auto-parameterized Shape Grammar (APSG): Finally the APSG rules are applied to construct the architectural structures. These structures are made out of the motifs by manipulating them in 3D space generating a surface, flat or curved. These rules are auto-parameterized. This means that the parameters of the rules will be calculated automatically according to the user requirements. The rules are based on Euclidean transformations including translation, rotation and scaling. In general, the more complex the structure the more rules are applied.

In the next section we discuss further the details of the methodology developed.

4 Islamic Geometric Motif-Based Structures

4.1 Generating the Motif

In accordance to the Shape Grammar definition, I is a regular polygon, e.g. triangle, quadrilateral, pentagon, hexagon, heptagon, octagon etc., in our grammar. V_T is the terminal shape (initial shape without the marker), V_M is the spatial marker which positions the successive assemblage of the shape by its given shape rule, example shown in Fig. 3. Lastly the shape rules, R , are the rules generated to construct n -fold geometry.

Fig. 3. Shape grammar notation

Analyzing IGP: As the IS in our system is changeable to an n -polygon, the category of IGP will be of n -fold geometry. From [23] many 4-, 5-, 6-, 8-, 10-, 12-fold patterns of varying complexity were analyzed, patterns containing both stars and polygons, to derive the grammar rules.

PSG Rules: The parameterized shape grammar rules were derived with reference to the three distinct features of IGP; symmetry, emergence and self-

similarity as discussed in the introduction. The symmetry and emergence features both generate the rules of translation and rotation. A star emerges by translating and rotating (from the centre) a single hexagon. It generates a symmetrical 6-pointed star with an order of symmetry 6. Scaling the hexagon proportionally retains the original state of the shape and is a cause for generating self-similarity in the derived pattern. Further, Boolean operations are also utilized to generate emergence as it produces new emergent designs.

Figure 4 shows the grammar rules generated from numerous patterns. Rule 1 (R1) translates in either x - or y -axis. Rule 2 (R2) rotates the shape with parameter r , where r is calculated as:

$$r = 360/n \quad (1)$$

As symmetry is a very important feature, to distinguish the n -fold in a motif, the user is restricted to the parameterization of this rule, which can only be computed by *equation 1*. For example to create a 4-fold motif, the angle of each fold has to be $1/4$ of 2π , to create a regular 4-fold motif. Hence the user would simply specify the n -fold. The scale transformation is rule 3, (R3), with parameter, m , and lastly the Boolean operation of uniting and intersecting with or without the IS, is rule 4 (R4). The grammar also includes an *Add* and *Duplicate* rule. By definition this is now a Parameterized Shape Grammar (PSG).

Fig. 4. PSG rules

4.2 Constructing Motif-Based Structures

The most common features in Islamic architecture are the lattice work which produces windows or arches, or more simply a wall, a column which supports a building and the striking feature that stands out, the dome. Deconstructing each feature we can derive the common shape grammar rules, i.e. translation, rotation and scale in all three axes, x -, y - and z -axis.

Algorithms: For a user to construct the structures of their required size, specific algorithms are applied to model them. Each algorithm has a number of equations associated with it.

Wall: As the shape grammar rules are designed to create n -fold geometry, with an IS of a regular polygon, both the square and hexagonal grids are used to tessellate the generated motifs. For the square grid it has a simple translation in the x - and y -axis, multiplying the number of motifs translating horizontally and vertically by their width (w) and height (h) respectively. The hexagonal grid however is slightly complex, the hexagons are positioned in a staggered fashion to produce a periodic pattern. The motif is translated horizontally T_x and vertically T_y by the following vectors, as shown in Fig. 5.

$$T_x = (w(n_x), 0) \quad (2)$$

$$T_y = (x(-1)^{n_y}, y) \quad (3)$$

where n_x and n_y denote the number of motifs used to tessellate in the x - and y -axis respectively.

Fig. 5. Translating the motifs to tessellate periodically, creating a wall

Column: The column is of cylindrical shape. Like the wall, the motif is tessellated vertically for the square grid, with the fixed translation, *equation 4* it is also translated in the z -axis, *equation 5*. This forms the radius of the column.

$$P_y = y_i(n_y) \quad (4)$$

$$r_c = w(n_x)/2\pi \quad (5)$$

To create the curvature of the column a rotational angle of the motif is calculated by dividing 360° by n_x (*equation 6*). To follow the periodic tessellation of the motifs to tessellate in the hexagonal grid another rotational rule is applied for the motifs to tessellate offset vertically, where the rotation is half of θ , μ (*equation 7*).

$$\theta = 360/n_x \quad (6)$$

$$\mu = 360/(2n_x) \quad (7)$$

This first creates a spiral of n_y motifs tessellated vertically offset, which are then combined and duplicated n_x times to tessellate horizontally. The following steps are used to construct the column, shown in Fig. 6:

- *Translate in the z -axis, this creates the radius of the column.*
- *Translate in the y -axis, to tessellate the motifs vertically.*
- *Rotate in the y -axis from the centre, for the motifs to tessellate offset (for the hexagonal grid)*
- *Combine, duplicate and rotate in the y -axis from the centre to form the complete column.*

Dome: There are various types of domes constructed in Islamic architecture. In this work the construction of a simple dome, a hemisphere, will be analysed. This will generate a base algorithm which can be varied in the future to construct different styles of domes.

The algorithm to construct the dome structure exhibits self-similarity. If we take the top view of a dome which is adorned with IGP, a star is generated at its apex by the geometric patterns. Using the design of the apex star of the dome as an initial starting point, translate, rotate and scale transformations can be applied to the motifs. The motifs however tessellate from a single radial point (centre $(0, 0, 0)$), to create periodicity. This generates radial symmetry within the structure.

To create the notion of self-similarity a method of proportional scaling is applied. This would in return scale the motifs to fill in the gaps proportionally and would also use the same number of motifs, which would create the central inner star. To scale the motifs the scale factor is defined:

$$s = (\pi(2r_s + y))/(n_w - y\pi) \quad (8)$$

Fig. 6. Translation and rotation of the motifs to tessellate periodically to construct a column

Fig. 7. Centre: The structure of the dome. Outside: Flowchart to construct a dome.

To form the dome, Fig. 7 (centre), a number of radii are calculated. Alongside r_s which calculates the radius of the initial star at the apex, the radius of the dome is also calculated, *equation. 9*. This radius is an extension of r_c as it is multiplied by the scale factor as well. The radius is formed by the sum of the number of levels in the dome which are constructed by the motifs.

$$r_D = (h(n_x/2)\pi)(s^{L-1}) \quad (9)$$

where L is the number of levels in the dome given by:

$$L = n_x/4 \quad (10)$$

The levels are calculated by a ratio. The relation between the levels of the dome and the number of star points, which are made up of n_x , is 1:4. This is due to the portion of the dome being covered by the motifs, which is only a quarter.

The dome is initially constructed with an initial angle which is calculated to generate the curvature of the dome, this positions the motifs with the offset rotation rule:

$$\cos(\alpha) = (2r_D^2 - r_s^2)/(2r_D^2) \quad (11)$$

The flowchart shown in Fig. 7, describes the steps taken to construct a dome, where N calculates the motifs tessellating vertically and M horizontally. The next section illustrates the results of how the methods for generating and constructing structures described above have been implemented within Autodesk Maya.

5 Results

As mentioned previously constructing a motif-based structure involves a three-stage process. The first stage is choosing the initial shape, which is an n -polygon. The second is applying the PSG rules to generate the motif. By adopting the rules of the PSG, various motifs can be generated. All the motifs are rendered as a 3D model, as a 3D motif model can be implemented in various uses, e.g. 3D printing, creating 3D motifs to use as building blocks to create tessellations or even simply 3D molds of the motif which can be used to mold different motifs. Finally the third stage is constructing the structure with the chosen motif applying the APSG rules. The 3D modelling tool to generate Islamic geometric motif-based structures was developed and implemented in Autodesk Maya using the Python scripting language, which is one of Maya's scripting languages.

Graphical User Interface (GUI): The 3D modelling tool has a GUI which provides the user with four elements. Firstly choosing the initial shape by entering a parameter n to generate an n -polygon. Secondly, a choice of grammar rules based on Fig. 4. Each rule has an input field for the user to choose their parameters. Thirdly, an option to tessellate the generated motif, either in a square or hexagonal grid, and lastly to produce a periodic tessellated architectural structure of their choice, a wall, a column or a dome.

Fig. 8. (a) Original motif derived from PSG rules (b) changing the translation parameter creates another original motif (c) changing the initial shape to a dodecagon and leaving all the rules the same creates another original motif (d) changing the initial shape and leaving out the last rule generates another original motif.

Fig. 9. Examples of the motifs PSG can create from 4- to 12-fold motifs.

Generating the motifs: To illustrate the methodology to generate the motifs a few examples are shown in Fig. 8. Fig. 8(a) is an example of how an original IGP can be generated by applying the PSG rules. If the translation parameter is changed in the previous example leaving the rest of the other rules and parameters the same, another original IGP can be derived as shown in Fig. 8(b). If we change the initial shape from a hexagon to a decagon it also generates an original motif (see Fig. 8(c)). And finally Fig. 8(d), removing a rule and changing the initial shape again produces an original motif. A set of 3D generated motifs are shown in Figure 9 generated from various initial shapes from a quadrilateral to a dodecagon.

Constructing the Structures: Using the generated motifs and applying them to the construction algorithms of their structures, one can construct motif-based structures. Fig. 10 illustrates how the three structural algorithms are applied to one of the generated motifs to create the structures. Fig. 10 shows renders of the three architectural structures where the motifs are tessellated either in the square (left) or hexagonal grid (right).

6 Conclusion

The work presented in this paper demonstrates a methodology to construct 3D Islamic geometric motif-based structures. A 3D modelling tool has been successfully implemented in Autodesk Maya to generate and construct both Islamic geometric motifs and motif-based structures.

Numerous IGP and architectural structures were analysed to adopt the repetitive and self-similarity structure of IGP. A number of algorithms were formed to create simple structures like a wall, column and a dome using the Shape grammar approach.

In this work the Parameterized Shape Grammar (PSG) has generated not only original Islamic geometric motifs but also formed new motifs. It has the advantage of using parameters contained within each rule, to allow flexibility in the grammar rules. The grammar is defined to generate n -fold geometry, with the flexibility of choosing the initial shape. Although the grammars generate the motifs and motif based-structures there are however some limitations.

Overlapping motifs is a special feature in IGP as it forms motifs that cannot be distinguished by a viewer when viewed for the first time. In this system, overlapping the motifs would create distortions in the final tessellated pattern. Also, the ability to tessellate the final motif in only two specific grids is a limitation on the variety of patterns produced.

Even with these limitations in place, the shape grammar approach has been used to generate numerous different n -fold motifs. The successful implementation of the two grammars in Maya Python has been successfully accomplished allowing the user to create a variety of Islamic geometric motifs, original and new, from the GUI and proceeding then it to using it to construct structures of the required size.

Fig. 10. Renders of structures. Left: Motif tessellated in the square grid. Right: Motif tessellated in the hexagonal grid.

7 Future Work

Although the results presented have demonstrated the effectiveness of the shape grammar approach, it is being further developed by improving the existing research methodology and generating a new 3D procedural mapping technique. The research methodology is being enhanced in the following ways:

Using more than one motif: This would allow the user to construct a tessellation/surface formed from more than one motif as IGP are known to be constructed by a number of different motifs to create both beauty and complexity.

Tessellations: Tessellating the motifs in more than two tessellating grids would introduce more variation. There are 17 different symmetry groups in 2D (The 17-Wallpaper groups) [1]. At the moment the motif is tessellated in either the square or hexagonal grid. Changing the tessellation would further generate emerging shapes, which can form a more complex pattern.

3D Mapping: This would allow the user to map the generated motifs automatically without manually placing them with given equations, onto any given mesh or surface. Hence this would provide a procedural mapping technique which will include the main characteristic of IGP, symmetry. The generated 3D Shape grammar would implement mathematical methods to procedurally map the objects, allowing the motifs to automatically transform easily i.e. to bend, mold, and shear, within a degree where the distinction of the generated motif will not be distorted.

In this paper we have effectively demonstrated a methodology for generating 3D Islamic geometric motif-based structures by applying the parametric shape grammar approach with the use of a single user changeable regular polygon. With the proposed future work we can provide a unique and highly flexible grammar and toolset for creating IGP-based surfaces/structures to any given user specification.

References

1. Abas, S. J., Salman, A.: Geometric and Grouptheoretic Methods for Computer Graphic Studies of Islamic Symmetric Patterns. *Comput. Graph. Forum.* 11(1), 43–53 (1992)
2. Jowers, I., Prats, M., Eissa, H., Lee, J. H.: A study of emergence in the generation of Islamic geometric patterns. In: 15th International Conference on Computer-Aided Architectural Design Research, pp. 39–48, OU, Hong Kong (2010)
3. Abas, S. J., Salman, A. S., Moustafa, A., Atiya, M.: Symmetries of Islamic geometrical patterns. World Scientific (1995).
4. Djibril, M. O., Thami, R. O. H.: Islamic geometrical patterns indexing and classification using discrete symmetry groups. *J. Comput. Cult. Herit.* 1(2), 1–14 (2008)
5. Kaplan, C.: Islamic star patterns from polygons in contact. In: *Proc. Graph. Interface*, pp. 177–185, ACM, Ontario (2005)
6. Riether, G., Baerlecken, D.: Digital Girih: A Digital Interpretation Of Islamic Architecture. *Int. J. Archit. Comput.* 10(1), 1–12 (2012)

7. Cromwell, P.: Islamic geometric designs from the Topkap Scroll I: unusual arrangements of stars, *J. Math. Arts*, (2010)
8. Aljamali, A.M.: Classification and Design of Islamic Geometric Patterns Using Computer Graphics. In: 2nd International Conference in Visualisation, pp. 253-258, IEEE, Barcelona (2009)
9. Kaplan, C., Salesin, D.: Islamic star patterns in absolute geometry. *ACM Trans. Graph.* 23(2), 97-119 (2004)
10. Tennant, R.: Medieval Islamic architecture, quasicrystals, and Penrose and Girih tiles: questions from the classroom. *Symmetry: Culture and Science*. 20, (2008)
11. Lu, P., Steinhardt, P.: Decagonal and Quasi-crystalline tilings in medieval Islamic architecture. *Science*. 315(5815), 1106-1110 (2007)
12. Cromwell, P.: A modular design system based on the Star and Cross pattern. *J. Math. Arts*. 6(1), 29-42 (2012)
13. Cromwell, P.: The search for quasi-periodicity in Islamic 5-fold ornament. *Math. Intell.* (2009)
14. Stiny, G., Gips, J.: Shape Grammars and the Generative Specification of Painting and Sculpture. *IFIP Congress*. 117 (1971)
15. Speller, T., Whitney, D., Crawley, E.: Using shape grammar to derive cellular automata rule patterns. *Complex Syst.* 79-297 (2007)
16. Cenani, S., Cagdas, G.: A shape grammar study: form generation with geometric Islamic patterns. In: 10th Generative Art International conference, pp. 1-7, Italy (2007)
17. Stiny, G., Mitchell, W.: The grammar of paradise: on the generation of Mughul gardens. *Environ. Plan. B*. 7(2), 209-226 (1980)
18. Rollo, J.: Triangle and T-square: the windows of Frank Lloyd Wright. *Environ. Plan. B*. 22(1), 75-92 (1995)
19. Ada, G.: A shape grammar: the language of traditional Turkish houses. *Environ. Plan. B*. 23(4), 443-464 (1996)
20. Stiny, G.: Ice-ray: a note on the generation of Chinese lattice designs. *Environ. Plan. B*. 4(1), 89-98 (1977)
21. Knight, T.: The generation of Hepplewhite-style chair-back designs. *Environ. Plan. B*. 7(2), 227-238 (1980)
22. Ebru, U.: A Shape Grammar Model To Generate Islamic Geometric Pattern. In: 12th Generative Art International conference, pp. 290-297, Italy (2009)
23. Broug, E.: Islamic Geometric Design. London: Thames and Hudson, London (2013)