xCoAx 2021 9th Conference on
Computation, Communication, Aesthetics & X

2021.xCoAx.org

Digital Doilies: Iterative Behavior as a Poetic Strategy

Keywords: Generative Art, Cellular Automata, Crochet Pattern, Poetics

André Silva
andrearqurb@outlook.com
Federal University of Minas Gerais, Belo Horizonte, Brazil

Marilia Bergamo
marilialb@ufmg.br
Federal University of Minas Gerais, Belo Horizonte, Brazil

Artists and researchers have widely explored the use of computational technology in artworks. This technology has shown the potential for aesthetic expansion when applied to other creative fields. For the perspective of computational creativity, the use of programming to explore traditional techniques within the craft practice appears to be enticing, due to its vast possibilities of exploration, which goes, for example, from woven fibers to sculpted artifacts. But in these practices the characteristic that stands out is the iterative process of craftsmanship. In this paper, we discuss the poetic practices that explore traditional techniques through computational approaches. Using a Cellular Automata algorithm that generates crochet lace patterns, we use the generative pattern and its materialization to illustrate and visualize the poetic practices that use repetition and reproduction as the base for their existence and survival.

1. Introduction

Most of the traditional techniques within the craft practice are based on iterative processes. Woven baskets, embroidery, pottery, and crochet, which is the case of study of this paper are a few examples to name. Originality and exclusivity were never a requirement regarding craftwork production. In their core reside a utilitarian function. But besides function, what differs craft from industrialization[1] is the capacity to combine beauty and purpose (Paz 1986). These artifacts represent local traditions, and they are incorporated into daily life. In general, the production comes from a family context or a small group of neighbors, which facilitates passing on knowledge, techniques, processes, and original designs. The importance and cultural value of these artifacts are related to the fact that they are the repository of the past, collectors of stories transmitted from generation to generation, and because they are an inseparable part of the community behaviors (Santana 2010). But beyond the reproduction and replication of existent patterns, which is a common characteristic among those techniques, they share an iterative process of craftsmanship intrinsic to their existence. In other words, the artifacts produced by iterative process of craftsmanship, have as their structure, reproduction, repetition, and replication values.

1. First Industrial Revolution.

These values are also to be found in computational processes, algorithm scripts, and so on and if both fields of knowledge share common characteristics, it seems reasonable to connect them. But this connection should be well-thought-out and not necessarily seek the total automation of the creative process, but its exploration through a method that allows participation and input from all the parts involved. In this paper, we present a Cellular Automata (CA) algorithm that generates crochet lace patterns. The CA as a mathematical approach uses the current result as a reference for the future generation of results, this characteristic is what makes it different from other traditional deterministic methods. Krawczyk points out that this recursive replacement method continues until some state is achieved (Krawczyk 2002). For the author, in parametric methods the results could be easily anticipated, which is not the case of recursive methods. This “out of control” behavior or algorithm creative autonomy opens up the possibilities for the crochet pattern exploration through an approach based on repetition, reproduction, and replication, which are values already consolidated in the traditional crochet practice. While the CA has some levels of creative autonomy, it is significant the participation from the other involved parties, such as, the lacemaker and artist. Those parties manipulate the CA logic inscribed in the algorithm, validate and select what is to be produced, and during the production phase, the lacemaker has the freedom to choose

the best path to materialize the pattern, since the algorithm does not produce the rules of materialization.

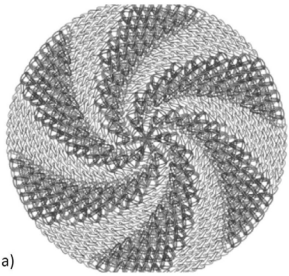
2. Algorithmic perspective

In terms of pattern making, crochet lace patterns are developed through actions that inscribe the materialization of crochet pieces. These actions are rules that dictate how the thread is involved by the crochet needle, types of crochet stitches, pattern modules, and how the pattern is built. Crochet patterns are algorithmic because conceptually they exist as a group of defined processes (Kenning 2007). This group of processes is the actions needed to materialize a crochet pattern. Crochet lace patterns are also iterative. The same group of processes could be replicated from the beginning to the end. Another characteristic found in crochet patterns is their ability to be replicated into the digital environment since they exist not only as physical forms but as code. At the moment that the verbal instructions are shortened, a syntax that includes modules and feedback loops is created. This syntax is similar to code writing in computation. The crochet pattern instructions behave as code to be interpreted by the lacemaker (Kenning 2007).

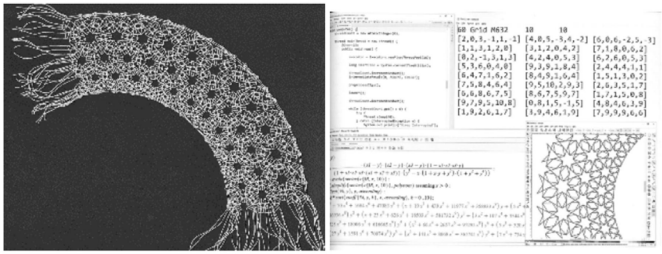
Still, according to Kenning, the digital environment offers several opportunities for this format of making patterns. The development of the pattern can become a hybrid of human and technological influences. The pattern can be influenced not only by the subjective decisions of the lacemaker, but can also be exposed to technological inputs (mouse, keyboard, etc.), and/or be influenced by the flow of information from the programming scripts and operating systems in which the pattern is immersed. The digital environment can handle complex algorithms and allows larger iterative processes with numerous patterns emergences (Kenning 2007). The last characteristic to consider regarding the development of crochet lace patterns in the digital environment is the extent to which patterns can be recognized when they are translated and transformed. It may not be easy to recognize emerging patterns due to our lack of experience with the evolved pattern formed. Thus, such explorations require an open mind when evaluating the pieces created (Kenning 2007).

A couple of examples of this algorithmic perspective found in the craftwork can be noticed in the work developed by Gail Kenning (2007) whose efforts focused on understanding the exploration of crochet patterns through an approach the replicates its final physical form, and the work by Veronika Irvine (2014) who developed an algorithm that generates bobbin lace patterns. In their work, they

designed algorithms that generate lace patterns ready for production. Even though Kenning focused on the digital development of patterns and not necessarily their materialization, the algorithm designed by her seems to produce "ready to go" patterns, which is the same case of Veronika's work, see figure 1. In our experiment, we use the cellular automata method that generates patterns with some level of abstraction that demand interpretation, validation, and some input from the lacemaker at the materialization phase. There are decision-making moments when the lacemaker is fabricating a pattern. For example, the lacemakers can choose how many stitches to connect to the previous row, they can choose to materialize each row from the start to the end or to see the rows as a unique spiral. The overall idea is to understand that what the algorithm produces is a suggestion and not a "ready to manufacture" pattern. Another difference in our experiment is the graphic pattern. We chose to work with symbols, because of that, it is only possible to see the translation of digital graphics to physical form if the pattern is materialized.



a)



b)

Fig. 1. a) Whirlpool design generated by an autonomous algorithm. Source: (Kenning 2007). b) Delle Caustiche work process by Veronika Irvine. Source: (Kanagy-Loux, Mills, and Neff 2019).

In figure 2.a., we can see a traditional pattern and instructions. In this scenario, any lacemaker who follows this material will materialize the same circular doily. In figure 2.b., we can see one graphic pattern generated by the CA and two different pieces. The first one (middle of figure 2.b.) is a materialization done

to be close as possible to the algorithm stitches distribution (top of figure 2.b.), the second one (bottom of figure 2.b.) is a piece with few alterations from the lacemaker which decided to connect loose groups of stitches to the previous row. The lack of information generated by the image purposely instigates the lacemaker's creativity because there are creative decisions to be made by the lacemaker. In this production method, the lacemaker and artist decides the algorithm behavior logic, the algorithm gives back an output in form of images, and then, again, the lacemaker materializes the final piece following his/her personal understanding of the output image generated by the code.

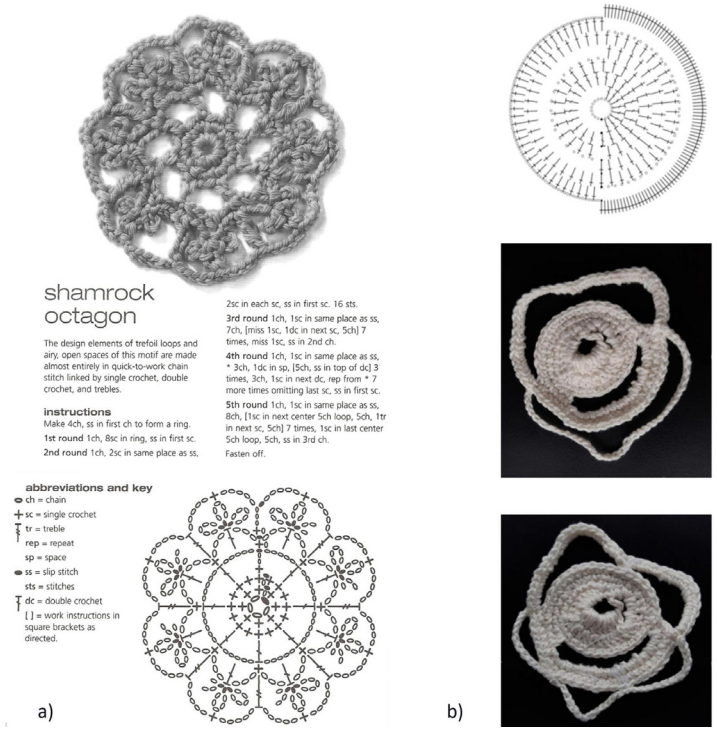


Fig. 2. a) Traditional pattern and instructions. Source: mimoedu.blogspot.
b) Pieces produced by computer and human influences. Source: The authors.

3. CA algorithm

The algorithm designed for this experiment works with four basic crochet stitches: Chain Stitch; a Slip Stitch to connect ends; Single Crochet and Double Crochet. In a physical crochet piece, the Chain Stitch is the base point to start a piece or change the direction of an existing one, and it can be used as an inde-

pendent element. The Slip Stitch makes the needle advance without letting the fabric get too high, due to this property it is commonly used for finishing circles and for finishing rows in circular pieces. It is also possible to make an entire row with this stitch around an edge for finishing, but it can not be used as an independent element. The Single crochet has proportional height and width. A Double Crochet is high enough to go over a row made of Single Crochet and reach the previous row, or leave a larger space.

The initial design of the algorithm is based on random behavior, this first experiment was important in order to understand the intrinsic positions necessary for a doily form and the dynamic potentials of the code. We fixed the parameters of distance from the first row, then the distance between rows, and the number of stitches per row to be used. We then established the limit of seven rows for the study of the graphic qualities of the images. As observed in figure 3.

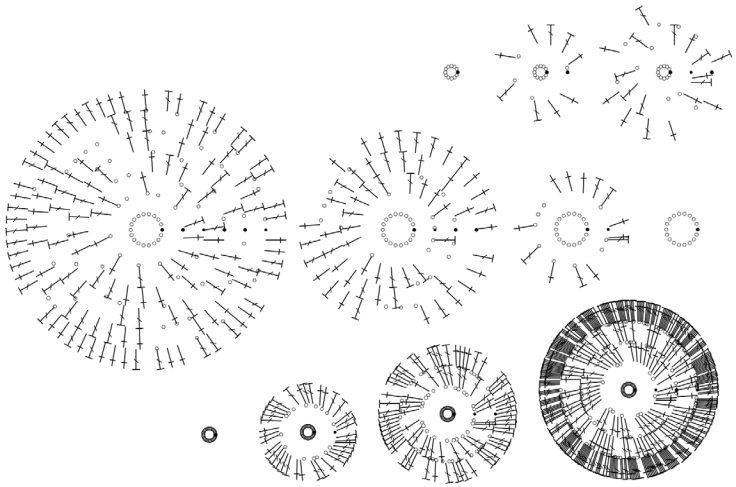


Fig. 3. Analysis of the random algorithm, in these images the central row was highlighted from the rest of the pieces for better visualization of the initial conditions of the system. Source: The authors.

Not all alternatives are likely to be physically implemented, but at this point of the work, there was a concern to investigate the potential of random values that could be altered for the generation of forms. Analyzing figure 3, it is possible to highlight one of the most basic characteristics of dynamic systems, the dependence on the initial conditions of the systems. If the system starts with a lot of amplitude between the stitches (at the top of the image) it becomes difficult to be produced, because the variation between the possible stitches and the distance between them creates many empty spaces. Likewise, when the distance between the stitches is very small (at the bottom of figure 3), the

number of stitches can become very overloaded, and it is necessary to investigate the ability to physically create a piece like this. In the center of the image, we have a possible initial condition of the system, to create final pieces with six rows. However, the random algorithm due to the very random nature of the exchanges does not generate discernible patterns of composition.

This initial design of random code was the foundation for our Cellular Automata model. From here we will explain the code written on processing. In the void setup () function, the values for the doilies are defined, such as, the number of circles and neighborhoods. Some of these parameters can be manipulated according to the interest of the lacemaker and artist.

```
void setup() {
  size(1300, 800);
  background(255);
  //define values
  circles = 8; //Number of circles on a dily
  e_steps = 20; //Cannot be less than 19,
or you have to use less circles
  e_radius = 38; //Initial radius on center
  e_gauge = 50; //Must change according to the size of images
  neighborhoods = 4; //Number of neighborhoods
  e_counter = 0; //Helps to count circles
  frameRate (5); //Slow frame rate to see pattern feneration
  drawCircles ();
}
```

Void drawCircles () is responsible for drawing the initial row of chain stitches and predefining the remaining rows. In the circles tab, we can find the script section responsible for arranging the points in the rows. It contains the random and the generative logic. In this section the class Circle [] and some variables are defined.

```
void drawCircles() { // MADE UP FUNCTION
  for (int i =0; i < circles; i++) {
    if (e_counter == 0) {
      //Create a first circle of points that cannot be changed by
any logic
      Circle thisCirc = new Circle(e_steps, e_radius, true, neigh-
borhoods);
      thisCirc.drawMe();
      _circleArr= (Circle[])append(_circleArr, thisCirc);
    } else if (e_counter < circles) {
      // Create all the other circles
      Circle thisCirc = new Circle(e_steps, e_radius, false, neigh-
borhoods);
      thisCirc.drawMe();
      _circleArr= (Circle[])append(_circleArr, thisCirc);
      e_radius += e_gauge;
      b_radious = e_radius - e_gauge;
      e_steps-=3;
    }
    e_counter++;
  }
}

class Circle {
  int steps;
  float radius;
  boolean first;
  int numberOfNeighborhoods;
  Point[]_pointArr = {}; // Array of Cellular Automata
  Circle(int s, float r, boolean f, int n) {
    steps = s;
    radius = r;
    first = f;
    numberOfNeighborhoods = n;
  }
```

The void updateMe() function contains the generative logic of the Cellular Automata model. The algorithm makes a random arrangement of the stitches and afterward, this function reads these stitches and starts to produce results according to the generative logic. This logic can be changed in this section.

```
void UpdateMe()

// Count the type of neighbors
        int[] thisNeighborhood = _pointArr[count]._myNeighbours;
        for (int x = 0; x <  thisNeighborhood.length; x++) {
          if (thisNeighborhood[x] == 0) {
            numberOfChainstitch++;
          } else if (thisNeighborhood[x] == 1) {
            numberOfSingleCrochet++;
          } else if (thisNeighborhood[x] == 2) {
            numberOfDoubleCrochet++;
          }
        }
        //println("Chain Stitch:" + numberOfChainstitch);
        //println("Single Crochet:" + numberOfSingleCrochet);
        //println("Double Crochet:" + numberOfDoubleCrochet);
        //Apply logic
        if (numberOfSingleCrochet+numberOfDoubleCrochet > 3*(_
pointArr[count]._myNeighbours.length)/4) {
          _pointArr[count].drawMe(0); // becomes Chain Stitch
        } else if (numberOfChainstitch<2*_pointArr[count]._myNeigh-
bours.length/3) {
          _pointArr[count].drawMe(1); // becomes Single Crochet
        } else if (numberOfChainstitch+numberOfSingleCrochet>2*_
pointArr[count]._myNeighbours.length/3) {
          _pointArr[count].drawMe(2); // becomes DoubleCrochet
        } else {
          //_pointArr[count].drawMe(int(random(3)));
```

Figure 4 shows two pieces generated by the random logic. This logic works by arranging random stitches through the circular grid. The patterns generated by this algorithm are generally denser, due to the lack of chain stitches that do not allow empty spaces in the physical piece. However, it was noticed that the few empty spaces found in the pieces occurred in the presence of chain stitches. It was observed that isolated chain stitches do not represent any considerable space, but from the set of 3 stitches, it is possible to observe considerable empty spaces.

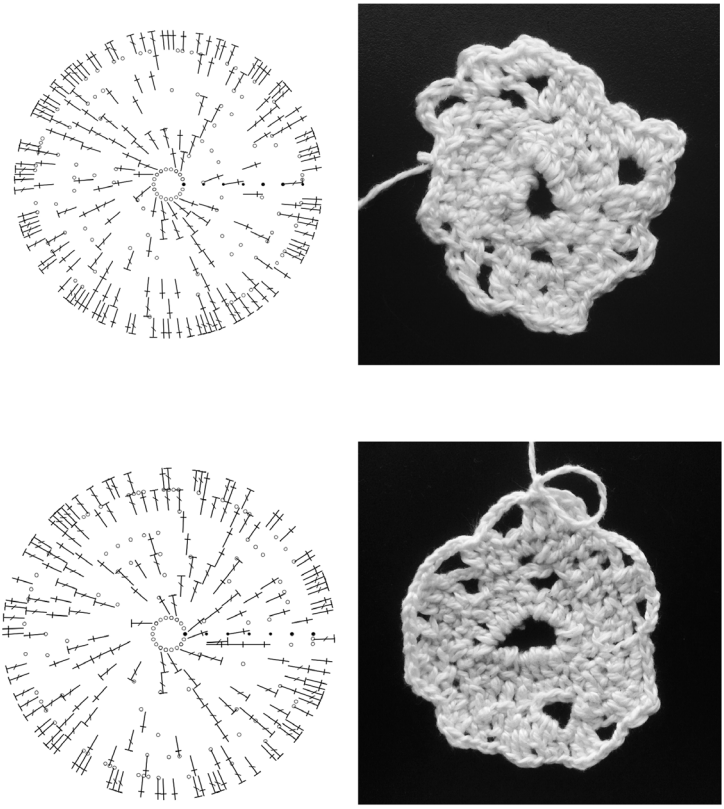


Fig. 4. Pieces generated by the random algorithm. Source: The authors.

For the second phase of the experiment, we worked with the CA method, which was explained previously. As the analysis of all stitches in the system requires a lot of computing resources, we established a logic that dialogs with the concept of the Game of Life by John Conway (Gardner 1970). We established a neighborhood where the stitch will observe the states of its neighbors and decide what it will become. This neighborhood can be parameterized by the code, and we can work with the entire piece as a single neighborhood, or by dividing the circular piece by the number of desired neighborhoods. Of course, there is a limit, a very high number of neighborhoods will not make sense for the division of parts of the doily. As in John Conway's Game of Life, stitches must choose one state, but we work with three states instead of two, a stitch can choose between being a chain stitch, single crochet, or double crochet, this decision depends on the logic below.

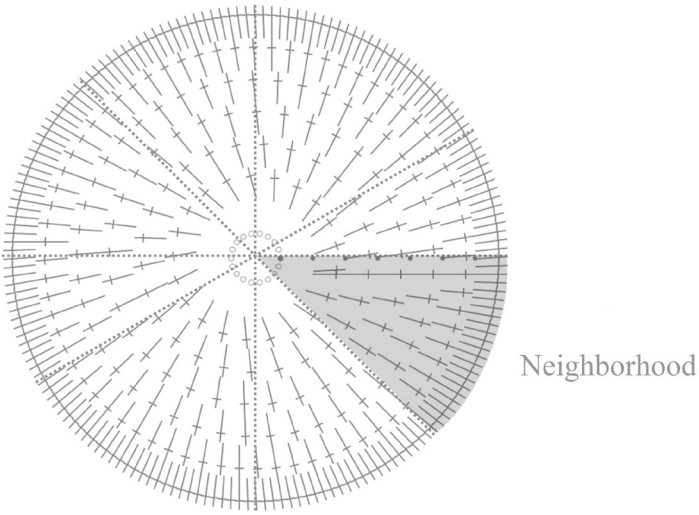


Fig. 5. Generative logic - Neighborhood concept. Source: The authors.

» Chain stitch: if the number of neighbors as Single Crochet or Double Crochet is greater than ¾ of the neighborhood;

» Single Crochet: if the number of Chain stitch is less than ⅓ of the neighborhood

» Double Crochet: if the sum of Chain stitch and Single Crochet is greater than ½ of the neighborhood;

The logic demonstrates the potential for generating patterns from the inclusion of computational creativity. In figure 6, the patterns generated for settings with one and two neighborhoods are shown. In the case of only one neighborhood (6a) the code always presents the same pattern for eight rows, which at the end of some life cycles always ends with the second row of double crochet, the third and the fourth of single crochet, and the fifth row of chain stitch, then again two rows of single crochet and one row of chain stitch. Still, in Figure 6, we have the generative behavior for two neighborhoods, this configuration always presented a total of four (6b, 6c, 6d, and 6e) patterns that alternated in the first neighborhood, while the second neighborhood was never changed. In figure 6 we present the result for the case of four neighborhoods, and again the last neighborhood remains unchanged, generating no pattern. In contrast, the first, second, and third quadrant showed different patterns than when there were only two neighborhoods (6f - first quadrant). But the first, second, and third

quadrant also showed similar patterns to what happened with only two neighborhoods (6g - first quadrant).

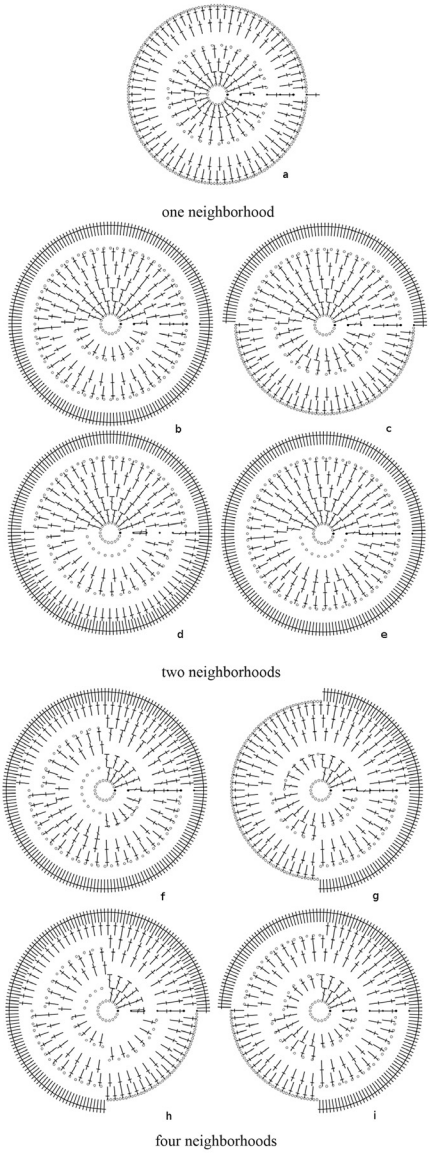


Fig. 6. Observed patterns for one, two and four neighborhoods. Source: The authors.

Despite the potential of the patterns found in figure 6, the purpose of this paper was not to create a series of observable patterns as in the case of the Game of Life but to validate the logic of the Cellular Automata as a potential for computational creativity and to visualize the poetics found in processes of repetition. Soon after this stage, instead of 1, 2, and 4 neighborhoods, it was decided to work with 2, 4, and 8. Some results can be seen in figure 7.

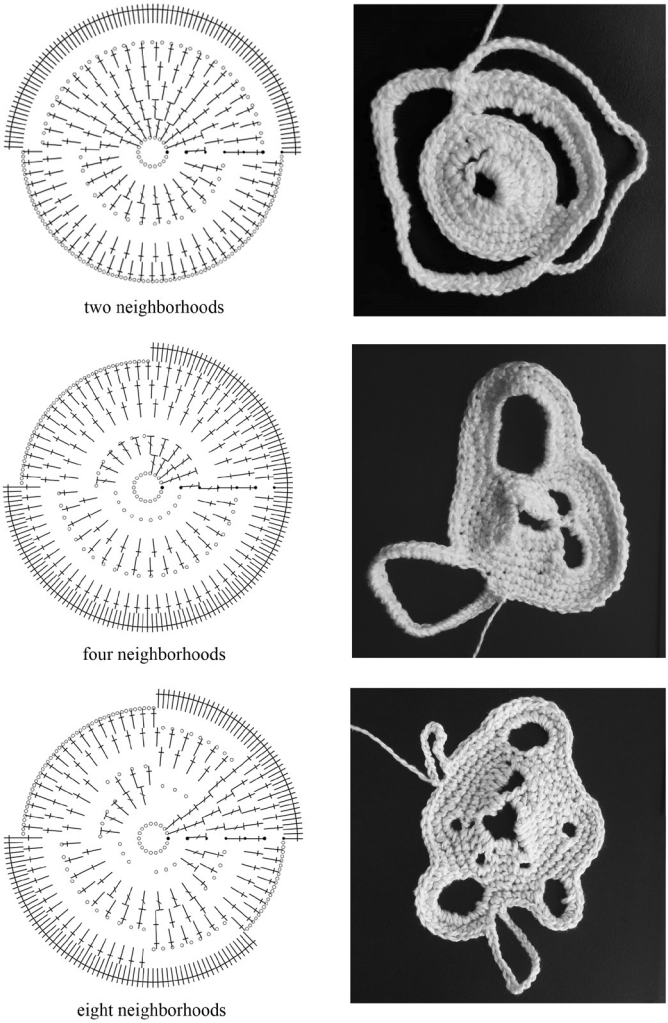


Fig. 7. Observed patterns for two, four, and eight neighborhoods. Source: The authors.

4. Poetics

Gilbert Simondon points out that it is the insertion that defines the aesthetic object and not the imitation (Simondon 2007). The philosopher makes this statement from the perspective of the industrial machine. However, craftsmanship precedes this industrial state and the massification of computing happens after it, and both use imitation, repetition, and reproduction as poetic values. The research object of this paper is inserted in a context in which the artisanal and computational technical objects use reproduction as a poetic and productive strategy. Both have functions whose understanding is necessary to comprehend completely their aesthetic value. Simondon points out that the discovery of the technical object's beauty cannot be left to perception alone. The function of the object must be understood and thought out. In other words, technical education is needed, so the beauty of technical objects can appear as the insertion of the technical schemes of a universe, at the key points of it. There is a lack of understanding of the object's function, which is necessary to correctly imagine and aesthetically perceive its structure and its relationship with the world (Simondon 2007).

The handicraft beauty is inseparable from function. Handicrafts are beautiful because they are useful, as they belong to a world before the separation between useful and beautiful (Paz 1986). The industrial object tends to disappear as a form, and become confused as a function. Its existence is its meaning, and its meaning is to be useful. Artworks are at the opposite side of industrial logic. Handicraft is a mediation, its forms are governed not only by the functionality but also by pleasure. The industrial object does not tolerate the superfluous, but the craftsmanship is satisfied in the adornments. Things are pleasant because they are useful and beautiful (Paz 1986).

The handmade object is not limited to its utilitarian function, it also reflects its decorative function. In the context of crochet, the decorative function prevails. Like a large part of craft production, the existence of crochet pieces occurs through the reproduction of existing patterns, which through small alterations from reproduction to reproduction, changes slowly. In addition to symmetry, shape, and color, we can see the poetics and beauty found in the detailed repetition, that within each reproduction, details are modified. This way the technique and the physical result can adapt to the needs and the environment where they are placed, guaranteeing survival and evolution of the technique, even in very small steps.

Understanding the crochet technique is crucial to comprehend its poetics. In addition to the pattern reproduction habit of the crochet practice, its materialization is an iterative process. The algorithm presented in this paper potentialized an intrinsic behavior of the crochet practice. The algorithm generates through repetition and similarity, but never identical patterns.

"Computational poetics take advantage of information compressed into small codes to generate otherness and beauty from generative processes" (Bergamo and Marinho 2019, 189). Crochet patterns behave this way in both physical and digital environments. The crochet piece is materialized through a group of processes, which are the necessary actions for the construction of a pattern, these processes are summarized in recipes that are interpreted by the lacemaker (Kenning 2007). In the digital environment, when translated into generative algorithms, these processes are even more compressed. The crochet practice has its iterative potential elevated when placed into a digital structure and poetics is found in that potential. Repetition and reproduction in biological systems guarantee the evolution of species, for the crochet technique, this potential grants the capacity of the emergence of novelty in processes that seem to never change or evolve.

5. Conclusion

The efforts of this paper were concentrated on explaining the algorithm designed to generate crochet lace patterns, the crochet technique, and the materialization of the patterns that emerged from the generative method. The understanding of these elements is necessary to perceive the poetics as we the authors see it. The constant and non-stop repetition allows for change and evolution, there is beauty and aesthetics in the subtle changes, and it is possible to get novelty from repetition. As highlighted in the previous sections, the crochet technique is an iterative process based on the repetition of actions and replication of existing patterns. The CA as a generative method potentialized the crochet iterative behavior. The CA works by repeating actions over and over. One cell needs to read its neighbors state to decide what it will become, this behavior is repeated until some desirable outcome is achieved. This behavior is analogous to the crochet technique because the lacemaker has to repeat modules of information to achieve a materialized piece. The generation of patterns by the CA simulated an evolutionary process based on repetition applied to crochet patterns. Even though the graphic patterns looked alike, when materialized, they presented unpredictable visuals. Simple stitches and behavior rules were enough to produce patterns that present a different aesthetic when compared to

the traditional practice. The emergence of unexpected results is what instigates the artist who works with generative approaches. In addition to the emergence of unexplored structures, we could perceive the subtlety and poetics presented in the repetition not only of the algorithm, but also, the lacemaker, who from generation to generation materialized different, mutated, evolved, transformed, but similar results.

Acknowledgements. Acknowledgement to CAPES for the financial assistance used to carry out this research.

References

Bergamo, Marília L. and Francisco Marinho. 2016. *Tecnologia e Delicadeza: estratégias da simplicidade cotidiana na geração de resultados estéticos complexos*. In: #15.ART Encontro Internacional de Arte e Tecnologia, 2016, Brasília. 15° Encontro Internacional de Arte e Tecnologia (#15.ART): arte, ação e participação, p. no Prelo.

Irvine, Veronika, and Frank Ruskey. 2014. *Developing a mathematical model for bobbin lace*. Journal of Mathematics and the Arts, 8:3-4, pp 95-110. DOI:10.1080/17513472.2014.982938.

Gardner, Martin. 1970. *The Fantastic Combinations of John Conway's New Solitaire Game of "Life"*. Scientific American, 223, pp. 120-123.

Kanagy-Loux, Elena, Amy Mills, and Nancy Neff. 2019. *Lace, not Lace: Contemporary Fiber Art from Lacemaking Techniques*. Hunterdon Art Museum, Clinton, New Jersey.

Kenning, Gail Joy. 2007. *Pattern as Process: An aesthetic exploration of the digital possibilities for conventional physical lace patterns*. Tese (Doutorado em Artes). University of New South Wales, Australia.

Krawczyk, Robert J. 2002. *Architectural Interpretation of Cellular Automata*. Generative Art.

Paz, Ottavio. 1996. *O uso e a contemplação*. In : História geral da Arte.Artes decorativas I. Artes decorativas aplicadas a habitação: desde o mundo antigo até o barroco. Ediciones Del Prado.

Santana, Maíra F. 2013. *Design e artesanato: fragilidades de uma aproximação*. Cadernos Gestão Social, 3(2),103-115.

Simondon, Gilbert. 2007. *Relaciones entre el pensamiento técnico y otras*. In G. Simondon.El modo de existencia de los objetos técnicos. Prometeo Libros Editorial.