Representing Crochet with Stitch Meshes

Runbo Guo
runbog@andrew.cmu.edu
Carnegie Mellon University

Vidya Narayanan
vidyan@cs.cmu.edu
Carnegie Mellon University

Jenny Lin
jennylin@cs.cmu.edu
Carnegie Mellon University

James McCann
jmccann@cs.cmu.edu
Carnegie Mellon University

Figure 1: In this paper, we describe how stitch meshes can be used to represent crochet patterns in a way which is amenable to yarn-level visualization and simulation.

ABSTRACT

Crochet is a fabrication technique in which a 3D surface is created from yarn by interlacing loops formed with a special hook. Crochet patterns are typically represented using a standardized set of abstract pictorial symbols. Unfortunately, while this notation is enough for someone well-versed in the individual stitches, it does not directly show the yarn layout of stitches. This lack of specification makes it difficult for both novice users and computer programs to parse, visualize, and design crochet patterns.

We demonstrate how to represent crochet patterns within the “stitch mesh” paradigm. That is, the pattern is represented using a library of tiles, where each tile contains yarn geometry, and tiles connect along their edges. In order to adapt stitch meshes to crochet, we introduce a special edge type which captures the idea of the *current loop* – the loop of yarn held on the crochet hook during fabrication. We also create a library of mesh face types which model commonly-used crochet stitches. We illustrate the richness of the crochet stitch faces by showing a number of examples including patterns generated from 3D models.

CCS CONCEPTS

• **Computing methodologies** → **Graphics systems and interfaces**; • **Applied computing** → **Computer-aided design**.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SCF '20, November 5–6, 2020, Virtual Event, USA

© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-8170-3/20/11...\$15.00
<https://doi.org/10.1145/3424630.3425409>

KEYWORDS

yarn, crochet, tiles, stitch mesh, yarn model, 3D crochet

ACM Reference Format:

Runbo Guo, Jenny Lin, Vidya Narayanan, and James McCann. 2020. Representing Crochet with Stitch Meshes. In *Symposium on Computational Fabrication (SCF '20)*, November 5–6, 2020, Virtual Event, USA. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/3424630.3425409>

1 INTRODUCTION

Crochet, like knitting, is a fabrication technique that involves manipulation of a continuous strand of yarn into inter-meshed loops in a way that forms a stable final object (Figure 1). However, the construction method produces a different final structure than knitting (Figure 2), and techniques to represent or design knitting patterns are not directly compatible with crochet patterns. Using just a few different styles of stitch construction, crocheters have created a variety of complex shapes [Forbes and Forbes 2007; Henderson 2001; Osinga and Krauskopf 2004]. To describe and share crochet patterns, hand-crafters use textual instructions or symbolic representations (e.g., based on the international crochet notation [Craft Yarn Council 2020]). Those familiar with crochet can easily read these patterns and understand the stitches and their construction order. However, these pattern formats typically do not describe how the stitches are constructed themselves or which stitch symbols are compatible with each other since they do not include any explicit information on how yarn is used within each stitch. This makes international crochet notation inconvenient for computer-based pattern authoring, simulation, or visualization.

Recently, new representations for knit objects have been introduced in the computer graphics community with an aim to support design, yarn simulation, and fabrication [Narayanan et al. 2019; Wu et al. 2019; Yuksel et al. 2012]. Although crochet and knitting differ

in how loops are held and in their basic building blocks, they are both techniques that construct fabric by building new loops through old loops. Therefore, it is tempting to consider a stitch-mesh like representation for crochet patterns. In this work, we show that crochet can indeed benefit from a stitch-mesh based representation and describe how crochet stitches can be viewed as a tilable set of faces.

We build upon the Craft Yarn Council’s international crochet notation [Craft Yarn Council 2020] using the construction order of crochet stitches and their dependencies to come up with a set of stitch-faces that can be tiled to form a complete pattern. We show that this crochet face-set can be used to both manually design crochet patterns using a general stitch meshes interface and semi-automatically generate patterns from 3D models.

2 RELATED WORK

Crochet and knitting are flexible fabrication techniques that can be used to produce a wide variety of geometries and textures. Researchers have showcased the power of textile crafts like crochet in illustrating complex objects such as hyperbolic surfaces [Bauermann and Taimina 2013; Henderson 2001; Kucukoglu and Colakoglu 2013; Osinga and Krauskopf 2004]. Soft fabricated crochet and knit surfaces form a convenient skin for soft robots and smart wearables [Okazaki et al. 2014; Suguitan and Hoffman 2018]. Additionally, they can be used to route tendons or have flexible regions with conductive or thermo-chromatic properties by simply using different yarns with these properties [Albaugh et al. 2019; King et al. 2018; Okazaki et al. 2014].

To assist in pattern generation for complex and asymmetric geometries, various design systems have been introduced to generate patterns from 3D models for fabrication by hand or machines. These design systems take as input a 3D model or segmented versions of a 3D model, generate rows of stitches by slicing up the model, and then combine them into yarn paths. However, these systems either offer limited editing capabilities [Çapunaman et al. 2017; Igarashi et al. 2008] or are restricted to knitting patterns [Mori and Igarashi 2007; Narayanan et al. 2018; Popescu et al. 2018; Wu et al. 2019].

The use of traditional chart symbols and string diagrams when constructing patterns for knitting can be seen in the work of Itoh and colleagues, who describe techniques to iteratively cover a 3D mesh with a graph that encodes the cloth [Funahashi et al. 1999; Suzuki et al. 2000]. This cloth shape is then pasted with knitting patterns based on traditional chart symbols for knitted structures.

Figure 2: Crochet and knitting form fabrics with different yarn structures, appearances, and behaviors.

These chart symbols are mapped to tiles of string diagrams that describe the yarn path on the cloth [Miyazaki et al. 1995].

More recently, Yuksel and colleagues introduced the general idea of Stitch Meshes, which uses modular tiles called stitch faces to represent knit objects [Yuksel et al. 2012]. Like the string diagrams used by Miyazaki et al. [1995], a stitch face contains the geometry of an individual stitch but with the requirement that edges are crossed by either the loop(s) of the stitch or the yarn(s) forming the stitch. These edge types give a natural constraint for tiling, allowing them to be tiled along a surface. Researchers have also looked at patterning algorithms and interactive editing tools that generate stitch meshes from 3D models for both hand and machine knitting [Narayanan et al. 2019; Wu et al. 2018].

In addition to the overall 3D shape of a knit or crochet object, the differences in yarn geometry of different stitches can introduce subtle texture variations and drastic variations to the overall geometry and sizing of the patterns. Researchers have looked at automatically fixing patterns to account for the variations introduced by textures [Hofmann et al. 2019], interactive systems to edit patterns with domain specific languages [Kaspar et al. 2019; Narayanan et al. 2019], as well as yarn-level simulation systems to capture the final deformed form of the fabric [Kaldor et al. 2008; Leaf et al. 2018].

Finally, easy visualization of instructions is an important component of fabrication. Researchers have looked at shape-aligned layout maps for stitch notations [Briar 2013] as well as augmenting digital records of the piece in a way that makes construction and communication easier [Rosner 2010; Rosner and Ryokai 2010].

3 BACKGROUND

Crochet, derived from the french term *croc* for *hook*, involves a single hand-held crochet hook that is used to hold a *leading loop*, to pick up loops from the in-progress piece, and to pull loops through each other. These actions are combined to form new stitches. The *free yarn* connects the new loop on the hook to the ball of yarn.

In Figure 3, we see the construction process for a *single crochet* stitch. Note how the hook is inserted into the previous row of the fabric to create a new loop from the free yarn. An additional loop is formed on the hook and pulled through the remaining loops until only a single loop remains. The remaining loop becomes the leading loop for the next stitch. By changing the number of loops made, which loops they are pulled through and the order in which these operations are performed, different types of stitches like the *double crochet* or *treble crochet* can be created. These stitches differ in yarn topology, dimensions, and textures from one another, and they can be combined in various ways in the final piece (Figure 4). What all these stitches have in common, however, is that they start and end with a single leading loop on the hook. The stacking order of loops, size of the holes, and stitches all play an integral role in manipulating the geometry and texture of the object, making it important these details are captured.

3.1 International Crochet Notation

The international crochet notation provides a compact symbol for each basic crochet stitch or operation (Figure 5). This style of representation is widely used within the crochet community [Craft Yarn Council 2020; Schapper 2012]. These symbols are laid out

Figure 3: Crochet terms and process. (a) At the start of any crochet stitch, the *hook* will be holding a *leading loop* and will grab the *free yarn* to make new loops. To make a *single crochet* stitch, (b) the hook goes through the previous row in the fabric to grab the free yarn, pulling it through the row but not the leading loop to (c) form a second loop that sits on the hook. The hook then (d) pulls the free yarn through both loops on the hook, to finish the stitch and create (e) a new leading loop.

Figure 4: Sample of different crochet stitch types. Note how different stitches can introduce different heights and textures

in charts that convey not just the crochet instructions, but also the shape of the resulting piece (Figure 1, 6). Although this works reasonably well for flat patterns, for complex three-dimensional patterns, visualizing the final result can be challenging. Secondly, since the symbols do not explicitly describe the yarn layout within the stitch, it can be challenging to visualize the order in which these stitches must be constructed.

Nevertheless, these notations do indicate that a repeatable set of stitch representations is valuable for conveying crochet patterns. We use these notations as a basis to come up with a new stitch-mesh-based representation for crochet stitches.

4 METHOD

To adapt stitch meshes to crochet, we must define edge labels that take into account crochet's unique fabrication method. Recall that while crochet has only a single leading loop on the hook that is used to form new loops, loops can also be pulled through any existing loops in the fabric. However, though the construction of a given crochet stitch can be incredibly complex, they all begin and end

Figure 5: The symbols used in the International crochet notation are simple to read and write, but assume prior knowledge on how to fabricate each stitch. (Figure from [Craft Yarn Council 2020], used with permission.)

Figure 6: Example crochet patterns using international crochet notation for (a) rectangular swatch (b) circular swatch. Yarn topology is not explicit in these representations.

Figure 7: Faces in our crochet stitch mesh representation use short-term (orange) and long-term (purple) edge types to capture the different dependencies encountered during crochet fabrication. The leading loop and the free yarn cross the short-term edges to capture the construction style of crochet, while the long term edges are crossed only by loops. The contents of the stitch body are specific to different stitch types.

with a singular leading loop which can be used with the free yarn to create additional stitches. This sequential construction of stitches gives rise to short-term dependencies between consecutive stitches. Meanwhile, the loop-through-fabric operation generates long-term dependencies between the current stitch and some previously made stitch. We can translate the above dependencies into the following edge labels (Figure 7):

- Next: Produces a leading loop and free yarn for the next stitch.
- Previous: Consumes a leading loop and free yarn from the previous stitch.
- Future: Produces a loop through which a future loop can be pulled through.
- Past: Consumes a loop from a stitch made in the past to form a loop for the current stitch.

Faces with these labels can be connected as previous-next and past-future pairs to cover a full surface.

To determine the body of a crochet stitch face, we first draw the knot diagram for the stitch. Next, we identify which parts of the knot diagram correspond to the contents of each labelled edge. Finally, the remaining portion of the knot diagram is used to define the geometry of the stitch body. The results of this process are shown in Table 1.

Of note is that there exist several minor variations on each tile. Aside from mirrored versions that account for construction direction, crochet stitch faces must also account for how a loop is pulled through the previous row. These variations impact how the stitches interact with each other and can give rise to new patterns. This can be thought of as the number of loops that are consumed by the past edge. For example, when pulling a loop through the previous row of the fabric in a single crochet stitch, typically the loop is not pulled through the center of a previously made loop, but through the hole that is bounded by the loops of a previous stitch. However, a valid single crochet stitch can be formed by pulling through the

Figure 8: Small variations in crochet stitches can have a large impact on the finished appearance. Here, the same rectangle is shown with two stitch types.

Figure 9: Chain stitches used to create the height needed to join rows of treble crochet stitches.

center of a previously made loop, and this the standard technique used for the first row of single crochet after a foundation row of chain stitches. Pulling through the center of a loop (or crocheting in the back stitch) gives a stylistic difference even though the overall shape is similar (rectangular patch), as seen in Figure 8.

Similarly, variations of the chain stitch that do not introduce future and past edges can be used to create additional height at the edge of the fabric without which the piece would scrunch up. This is shown with a treble stitch rectangle in Figure 9.

In order to produce shapes that do not rest on a plane, apart from the standard stitches, one would need stitches that increase and decrease the number of stitches within a row as well as a way to create partial width rows. Decreasing the number of stitches can be accomplished with decrease tiles such as `sc2tog` and `sc3tog`, shown in Table 1. The `sc2tog` stitch uses two past loop edges, making a new stitch through both of them producing a single future loop edge. Similarly, `sc3tog` decreases through three past edges. In order to increase the number of stitches, multiple new stitches are formed through the same past stitch. In our stitch mesh setup, this is done using the `inc` utility face that, effectively, splits a future

Standard Stitches:				
Chain Stitch (ch) – used in starting row and to add slack between rows.

Single Crochet (sc) – basic building block.

Single Crochet N Together (scN tog) – narrows the fabric.

Utility Faces:				
Chain Turn (ch_edge{1, 2}) adds slack while turning yarn.		Turn (turn) turns between short rows.	Cap (cap) finishes a column.	Increase (inc) splits a loop to widen fabric.

Table 1: International crochet symbols, knot diagrams, and the corresponding stitch mesh faces used in this work. Not shown, but also used in our system, are mirrored and single-loop-input variants of these stitch mesh faces.

edge (produced loops) into two future edges. Creating partial-width rows is done using the turn face to crochet in the other direction before completing the entire row. To reduce the number of overall stitch mesh faces, we introduce a few more utility tiles such as the cap face and the turn face `ch_edge`. These utility faces can be added to any stitch face (along compatible edges) to produce minor variants without needing to create multiple such faces for each stitch type.

Although Table 1 only covers a subset of potential crochet stitches, these are already sufficient to generate a wide variety of crochet objects. Next, we describe a variety of crochet results both simulated and fabricated, generated using this tile set.

5 RESULTS

We used our set of crochet faces to represent a variety of crochet objects. All crochet faces and a selection of stitch meshes were made in a generic stitch mesh interface (Figure 10), while the remainder were generated using the pipeline from Narayanan et al. [2019] to generate stitch meshes from triangle meshes. The automatic

Figure 10: Except where noted otherwise, crochet stitch meshes in this paper were created using a generic stitch mesh interface which allows, (a) face-level editing and, (b) mesh creation/labelling.

Figure 11: Crochet stitch mesh patterns for a cube, a sphere, and the Stanford bunny (top) and their yarn-relaxed renderings (bottom). For the crochet-box pattern, we constructed a cube stitch mesh directly using our editing interface. To construct the sphere and Stanford Bunny, we first generated a stitch mesh representation from triangle mesh, and then placed the appropriate stitch-types using our interface.

pipeline takes a manifold triangulated 3D mesh with boundaries as input and generates a stitch mesh consisting of quad faces (mapped to single crochet sc or turn faces), pentagons (mapped to sc2tog), and increase triangles (mapped to inc). These are only a subset of the possible crochet faces and further edits can be applied using the interface. Automatically constructing suitable stitch meshes taking crochet patterning styles into account would be interesting future work. The stitch mesh interface was then used to set the face types to the appropriate crochet analogues.

In Figure 1, we see a basic pattern – a rectangle formed of only single crochet and chain stitches. When the yarns extracted from the stitch mesh structure are relaxed, it takes on a crochet-like appearance as shown by the physical pattern.

The example cube (Figure 11, left) and sphere (Figure 11, center) use both increase and decrease stitches for shaping. Specifically, the cube uses the more dramatically-decreasing “single crochet three together” stitch to attain the sharper corners, while the sphere uses the more gradual “single crochet two together” stitch. While the sides of the cube and its opening are a bit misshapen in the simulated result, the sharp corners help differentiate it from the sphere.

The Stanford bunny example (Figure 11, right) combines the increase and decrease faces to match the shape of the input mesh.

Of note is that due to the remeshing technique used, the stitch density is lower in the body region than the ears. This results in a more ‘lacy’ fabric with larger holes and less detailed shaping in the simulation compared the starting stitch mesh. Nonetheless, the crochet-like appearance of the pattern is maintained.

Finally, we generated both a stitch mesh and text instructions from the stitch mesh for the letters S, C, and F for hand fabrication. These shapes employ both increases and decreases, as well as turn utility faces to incorporate short-rows into the pattern. Figure 12 shows a comparison of the fabricated final results and the simulated patterns. The results appear similar, but do highlight some shortcomings of our current ad-hoc yarn-relaxation / simulation approach (dense stitches and large gaps in the simulated “S”); as well as the stitch-mesh generation approach (some small holes and non-ideal shaping in the hand-fabricated letters).

6 DISCUSSION

Stitch (Face) Modeling. In this work, we introduced a generic framework to construct crochet stitch meshes and introduced a tile-set of common crochet stitch faces. We showed that by using these faces, a wide variety of 3D shapes can be represented. However, this tile set is far from complete. Crochet is a complex craft and provides a lot of flexibility to the designer for constructing stitches,

Figure 12: The letters “SCF” translated into a crochet stitch mesh and (top) rendered with after yarn relaxation as well as (bottom) hand crocheted. The shaping of the bar in ‘F’ is not adequately captured by the automatically generated pattern and is nearly missing in the hand crocheted version.

and extending the tile set to cover more stitch types would make this design system more useful. Going back to the international crochet symbols, only the chain stitch, slip stitch, single crochet, sc2tog, and sc3tog were modeled by our method. It would not be unreasonable to extend it to the double crochet or the treble crochet stitch, which can be seen as a taller single stitch. Along with that, stitches such as the chain three picot can be modeled by combining already-made stitches together.

In addition to these stitches, some variations rely on working on a part of the fabric which is not captured by *future* edges. An example is the front-post-double-crochet stitch shown in Figure 13, which uses two adjacent holes and the loop between them to anchor the new loop to the fabric. Modeling such stitches with our framework would require special consideration and increase the stitch modeling effort for the designer.

As shown in Figures 8 and 13, minor variations to stitch types can create subtle but important textural differences to the final object. Creating all these variations by hand introduces additional modeling burdens. An automatic pipeline that generates all legal variants derived from a given stitch face would greatly streamline the process.

Pattern Modeling. To generate crochet stitch meshes for 3D shapes, our system uses a variant of the knit pattern generation system presented by Narayanan et al. [2019]. The face types are modified to match the stitch faces shown in Table 1, but this system uses a single stitch ‘aspect ratio’ to drive the remeshing process. Clearly, the different types of stitches, multiple yarns, and even the crocheter’s tensioning style, can introduce faces with different aspect ratios and this needs to be accounted for. Note the gaps introduced in the back of the Stanford Bunny pattern, for example (Figure 11). Remeshing algorithms can often generate poor patterns, especially around sharp features; notice the poor shaping of the shape ‘F’ in

Figure 13: The front post double crochet is an example of a stitch that cannot be represented by our current system. Here, the yarn enters into the piece within the internal stitch body, instead of the outer boundary as shown with the bottom left. This method of adding stitches creates a raised structure as shown to the right

Figure 12. A pattern-aware remeshing algorithm that takes these variations into consideration would be able to produce patterns with much higher accuracy. Incorporating better yarn-simulation systems within the design system can also help with pattern design, giving high-fidelity results before investing the time to hand-fabricate these complex patterns. Furthermore, the large amount of flexibility afforded to crochet as a fabrication technique raises the question of what, if any, limitations there are on crochetable structures. While the insights from Wu et al. [2019] for making stitch meshes knittable can be directly applied to create crochet meshes that can be reasonably fabricated, crochet has many more techniques than knitting that can resolve conflicting row directions or introduce local shaping. Codifying what constraints should still apply to crochet is an exciting area for future research. Finally, to fabricate these crochet patterns, we generate text-based instructions from the crochet stitch mesh, that the hand-crafter can follow along with the 3D stitch mesh. Introducing better interface tools and incorporating construction instructions as shown in Knittable Stitch Meshes [2019] may also improve the overall user experience for fabrication.

7 CONCLUSION

We present a new way of representing crochet patterns that uses a stitch mesh (with new face types) to explicitly define the yarn geometry. This simple, intuitive data structure can be used as the basis for creating visualizations and physical simulations; supporting computer-aided understanding; and design of crochet patterns. Further, such visualizations and simulations could provide a foundation for future crochet CAD tools for users of all skill levels.

ACKNOWLEDGMENTS

Thank you to Ella Moore for crochet expertise and for creating the “SCF” example from computer-generated instructions. Physical patterns used to illustrate the crochet process were based on “The Crochet Stitch Bible” and “The Complete Book of Crochet Stitch Designs: 500 Classic & Original Patterns”.

REFERENCES

- Lea Albaugh, Scott Hudson, and Lining Yao. 2019. Digital Fabrication of Soft Actuated Objects by Machine Knitting. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) (CHI '19). Association for Computing Machinery, New York, NY, USA, 1–13. <https://doi.org/10.1145/3290605.3300414>
- Gisela Baumann and Daina Taimina. 2013. Crocheting algorithms. *Cornell J. Archit* 9 (2013), 105–112.
- JC Briar. 2013. Stitch Maps. <https://stitch-maps.com/>.
- Özgüç Bertuğ Çapınanman, Cemal Koray Bingöl, and Benay Gürsoy. 2017. Computing Stitches and Crocheting Geometry. In *Computer-Aided Architectural Design. Future Trajectories*, Gülen Çağdaş, Mine Özkar, Leman Figen Gül, and Ethem Güler (Eds.). Springer Singapore, Singapore, 289–305.
- Craft Yarn Council. 2020. Crochet Chart Symbols. <https://media.craftyarncouncil.com/standards/crochet-chart-symbols>
- Jessica Forbes and Casey Forbes. 2007. Ravelry. <http://www.ravelry.com/>
- Tatsushi Funahashi, Masashi Yamada, Hirohisa Seki, and Hidenori Itoh. 1999. A Technique for Representing Cloth Shapes and Generating 3-Dimensional Knitting Shapes. *FORMA-TOKYO*-14, 3 (1999), 239–248.
- David W Henderson. 2001. Crocheting the hyperbolic plane. *Mathematical Intelligencer* 23, 2 (2001), 17–27.
- Megan Hofmann, Lea Albaugh, Ticha Sethapakadi, Jessica Hodgins, Scott E. Hudson, James McCann, and Jennifer Mankoff. 2019. KnitPicking Textures: Programming and Modifying Complex Knitted Textures for Machine and Hand Knitting. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) (UIST '19). Association for Computing Machinery, New York, NY, USA, 5–16. <https://doi.org/10.1145/3332165.3347886>
- Yuki Igarashi, Takeo Igarashi, and Hiromasa Suzuki. 2008. Knitty: 3D modeling of knitted animals with a production assistant interface. In *Eurographics 2008 Annex to the Conference Proceedings*.
- Jonathan M. Kaldor, Doug L. James, and Steve Marschner. 2008. Simulating Knitted Cloth at the Yarn Level. In *ACM SIGGRAPH 2008 Papers* (Los Angeles, California) (SIGGRAPH '08). Association for Computing Machinery, New York, NY, USA, Article 65, 9 pages. <https://doi.org/10.1145/1399504.1360664>
- Alexandre Kaspar, Liane Makatura, and Wojciech Matusik. 2019. Knitting Skeletons: Computer-Aided Design Tool for Shaping and Patterning of Knitted Garments. *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*.
- Jonathan P King, Dominik Bauer, Cornelia Schlangenhauf, Kai-Hung Chang, Daniele Moro, Nancy Pollard, and Stelian Coros. 2018. Design, fabrication, and evaluation of tendon-driven multi-fingered foam hands. In *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*. IEEE, 1–9.
- J. Gozde Kucukoglu and Birgul Colakoglu. 2013. Algorithmic Form Generation for Crochet Technique A study for decoding crocheted surface behaviour to explore variations. In *eCAADe 2013: Computation and Performance*, Vol. 2.
- Jonathan Leaf, Rundong Wu, Eston Schweickart, Doug L. James, and Steve Marschner. 2018. Interactive Design of Yarn-Level Cloth Patterns. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2018)* 37, 6 (11 2018). <https://doi.org/10.1145/3272127.3275105>
- T Miyazaki, Y Shimajiri, M Yamada, H Seki, and H Itoh. 1995. A knitting pattern recognition and stitch symbol generating system for knit designing. *Computers & Industrial Engineering* 29, 1-4 (1995), 669–673.
- Yuki Mori and Takeo Igarashi. 2007. Plushie: An Interactive Design System for Plush Toys. *ACM Trans. Graph.* 26, 3 (July 2007), 45–es. <https://doi.org/10.1145/1276377.1276433>
- Vidya Narayanan, Lea Albaugh, Jessica Hodgins, Stelian Coros, and James McCann. 2018. Automatic Machine Knitting of 3D Meshes. *ACM Trans. Graph.* 37, 3, Article 35 (Aug. 2018), 15 pages. <https://doi.org/10.1145/3186265>
- Vidya Narayanan, Kui Wu, Cem Yuksel, and James McCann. 2019. Visual knitting machine programming. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–13.
- Momoko Okazaki, Ken Nakagaki, and Yasuaki Kakehi. 2014. MetamoCrochet: Augmenting Crocheting with Bi-Stable Color Changing Inks. In *ACM SIGGRAPH 2014 Posters* (Vancouver, Canada) (SIGGRAPH '14). Association for Computing Machinery, New York, NY, USA, Article 19, 1 pages. <https://doi.org/10.1145/2614217.2633391>
- Hinke M Osinga and Bernd Krauskopf. 2004. Crocheting the Lorenz manifold. *Mathematical Intelligencer* 26, 4 (2004), 25–37.
- Mariana Popescu, Matthias Rippmann, Tom Van Mele, and Philippe Block. 2018. *Automated Generation of Knit Patterns for Non-developable Surfaces*. Springer Singapore, Singapore, 271–284. https://doi.org/10.1007/978-981-10-6611-5_24
- Daniela K Rosner. 2010. Mediated crafts: digital practices around creative handwork. In *CHI'10 Extended Abstracts on Human Factors in Computing Systems*. 2955–2958.
- Daniela K. Rosner and Kimiko Ryokai. 2010. Spyn: Augmenting the Creative and Communicative Potential of Craft. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) (CHI '10). Association for Computing Machinery, New York, NY, USA, 2407–2416. <https://doi.org/10.1145/1753326.1753691>
- Linda Schapper. 2012. *The complete book of crochet stitch designs: 500 classic & original patterns*. Lark.
- Michael Suguitan and Guy Hoffman. 2018. Blossom: A Tensile Social Robot Design with a Handcrafted Shell. In *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction* (Chicago, IL, USA) (HRI '18). Association for Computing Machinery, New York, NY, USA, 249–250. <https://doi.org/10.1145/3173386.3177019>
- Daisuke Suzuki, Tsuyoshi Nakamura, Lifeng He, and Hidenori Itoh. 2000. A supporting system for colored knitting design. *IEEE Transactions on Electronics, Information and Systems* 120, 12 (2000), 1833–1839.
- Kui Wu, Xifeng Gao, Zachary Ferguson, Daniele Panozzo, and Cem Yuksel. 2018. Stitch Meshing. *ACM Trans. Graph. (Proceedings of SIGGRAPH 2018)* 37, 4 (jul 2018), 130:1–130:14.
- Kui Wu, Hannah Swan, and Cem Yuksel. 2019. Knittable Stitch Meshes. *ACM Trans. Graph.* 38, 1 (Jan. 2019), 10:1–10:13.
- Cem Yuksel, Jonathan M. Kaldor, Doug L. James, and Steve Marschner. 2012. Stitch Meshes for Modeling Knitted Clothing with Yarn-level Detail. *ACM Trans. Graph. (Proceedings of SIGGRAPH 2012)* 31, 3 (2012), 37:1–37:12.