

Scan to Knit – From Body Scan Directly to the Knitting Machine

Dominik ŠURC*¹, Dominik MICHEL², Alexander MIROSNICENKO¹,
Alexander ARTSCHWAGER¹, Uwe RÖDER¹

¹ DITF Denkendorf, Denkendorf, Germany;

² Avalution GmbH, Kaiserslautern, Germany

<https://doi.org/10.15221/20.30>

Abstract

In this work, we describe an integrated solution to manufacture individualized made-to-measure (MtM) garments based on 3D body scanning by means of automatic data transmission from scanner to the flat knitting machine. The underlying process automatization reduces the complexity of the production of individualized lot-size-one garments.

The process consists of two main steps: from scan to garment order files and from garment order files to Jacquard patterns and subsequent knitting process.

Initially, a person is scanned with a 3D body scanner and body measures are extracted by the scanning software. After uploading the obtained body measures, the best fitting size is automatically suggested from size charts and corresponding fit rules for the selected garment type. The MtM Plugin of the scanning software is responsible for evaluating individual alteration rules of selected measures for the best fit. The resulting order file thus contains references to the garment piece file in the size chart database, the basic size and the alteration values. The second step comprises the algorithmic interpretation of 2D garment pieces and subsequent generation of Jacquard patterns. In addition to the piece file, information such as the material parameters is provided to the algorithm. The procedure starts with sampling of the garment piece according to the stitch construction, followed by piece manipulation to insure knitability. The resulting Jacquard pattern can be produced immediately by an appropriately configured flat knitting machine.

Keywords: flat knitting, knitting machine, made-to-measure garments, 3D body scanning, individualized textiles, digital manufacturing

1. Introduction

The integrated digitalization of all the manufacturing processes from product development to production is the key for successful digital transformation of the textile industry.

Nearshoring, the customers' requirement for more individually configured products puts the manufacturers under increased pressure. Short production cycles, small lot sizes and simultaneously increased product variety make the horizontal and vertical textile supply chains more complex and more difficult to handle. Although the flat knitting Jacquard technique offers individualization opportunities, it is only predefined and optimized for a continuous production of mainly homogenous fabrics. While the machines and production sites are highly efficient, the machine setup is time consuming. The process is therefore laid-out for high output and continuity.

In order to manufacture flat knit individualized garments as part of an MtM process, a high effort in the data preparation and processing is necessary. This includes the recording of product related data by 3D scanning the customer, the creation of corresponding MtM garment pieces and the implementation of the knitting programs. In extreme cases, the process needs to be done for each individual customer and for each garment piece separately.

We address these shortcomings by developing a workflow which can enable a flexible production of individualized knitwear based on customers' 3D body scan. We also highlight the advantages of individualized textiles and digital manufacturing for the apparel industry.

In the following, we first provide a short overview of the discussed topics. We describe in detail the scan-to-knit workflow and illustrate its components and first test results.

* dominik.surc@ditf.de; +49 711 9340 278; www.ditf.de

1.1. Made to Measure Garments

Made-to-measure (MtM) typically refers to customized clothing that is cut and sewn using a standard-sized base pattern with individualized adaptations to the body shape of the customer. Made-to-measure garments always involve some form of standardization in the pattern and manufacturing, as alteration rules describe the change of the pattern (along a few measurement lines) whenever a few body measurements of the customer deviate from the base pattern in a standard size. For this reason, common expectations are for MtM products to be more expensive than ready-to-wear garments, but provide far better fit to the customers' body. Nevertheless, as the customers' requirement for more individually configured products rises, the advancing digitalization offers a new angle for made-to-measure solutions.

Examples for MtM garments are mostly suits and sport coats. But every pattern can actually be prepared for MtM alterations, as long as the necessary transformation rules and methods between the standard sizes are provided. Therefore, not only cut & sew can be covered, but knitwear, too.

1.2. Flat Knitting

Knitting of whole garments piece without sewing – seamless flat knitting – has been developed by both leading flat knitting machine manufacturers Shima Seiki [1] and Stoll [2]. While these solutions provide a fast approach to a complete garment piece, they are not suited for individualized lot-size-one products and cannot be flexibly combined with garment CAD and simulation software.

Automatic generation of knitting patterns for flat knitting machines from geometric models has been an active research topic. Igarashi et al. [3,4] describe a pattern generation method for hand knit toys. Approaches for machine knitting focused on architecture appear in Popescu et al. [5] and Liu et al. [6]. A similar method tested on large variety of meshes is presented in McCann et al. [7] and in Narayanan et al. [8,9]. While these approaches demonstrate knitting pattern generation from 3D triangular surface meshes, some concepts can also be applied in 2D, particularly those regarding the manipulation of 2D structures to ensure knitability.

2. Method

This section presents the scan2knit workflow in detail, starting with the 3D scanning and the extraction of body measures. We outline the process leading to the order file and 2D garment pieces. Lastly, we describe the algorithm which generates knitting patterns from 2D garment pieces.

2.1. Scanning and Extraction of Body Measures

For a good fit, precise imaging and modelling of the human body are necessary. The human body in motion makes the measurement process more difficult and places considerable demands on the scanning time. High mechanical effort is necessary to accelerate the scanning process. The use of a laser light line full body scanner is therefore recommended. 3D scanning serves as a basis for the creation of avatars with homologous, invariant meshes which can be used to evaluate semantic and topological information for the extraction of body dimensions.

Alternatively, if a full body scanner is not available or not desired due to individual imaging, a virtual scanner can be used. This combines a small amount of personal data with statistical survey measurement knowledge about the German population to generate a mean avatar that best matches the input. Here, the main goal of the human virtualization is the extraction of body measurements. Knowledge of the exact body measures is needed in the next workflow step. The generated scanning avatars can be segmented into the body parts necessary for the fashion garments under consideration. The necessary body measurements for processing the garment can also be extracted from the avatar. For made-to-measure, the scanning has to be as precise as possible, making full body laser scanning mandatory. In case of this not being possible or feasible, other means of human body virtualization can be used. All scanning options result in a standardized mesh representation of the customer from which individual body measurements are extracted. There is a list of ISO-conform body measurements already extracted. In case of the MtM garment requiring a new body measure (like the lower arm girth two thirds of the way between elbow and wrist often used in compression garments), this measuring rule can be defined on the invariant, homologous mesh the avatar is based upon.

Figure 1 illustrates the first step in the scan-to-knit workflow, from scanning with avatar generation to body measurement extraction.

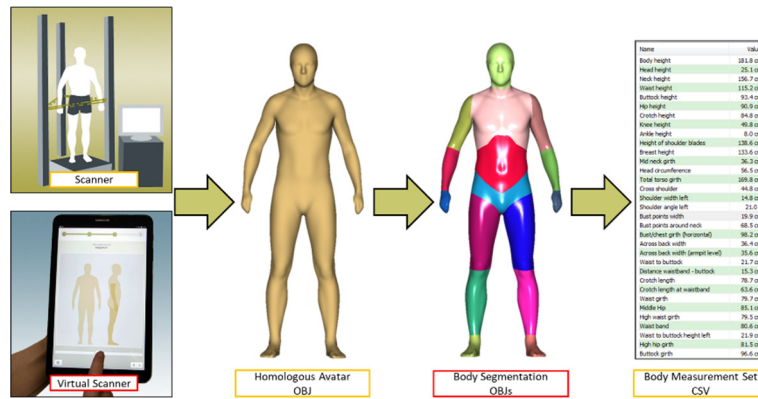


Fig. 1: Workflow component – from scanning to body measurements

For an MtM garment there is a defined set of body measurements required for alteration applications. At first, a standard size is determined. This is done by size recommendation, comparing the body measurements of the customer with the garment size chart and returning the closest matching size. Based on the base size, alteration values are calculated, being the difference between the size reference values and the body measurements of the customer.

The pattern has alteration rules included, describing its change based on alteration values. This transformation adapts the pattern from standard sizes to the individual features of the customer and represents the MtM changes.

Figure 2 illustrates the second step in the scan-to-knit workflow, from the garment and its pattern, combined with alteration values to an order file ready to be imported into the CAD-application of Vidya.

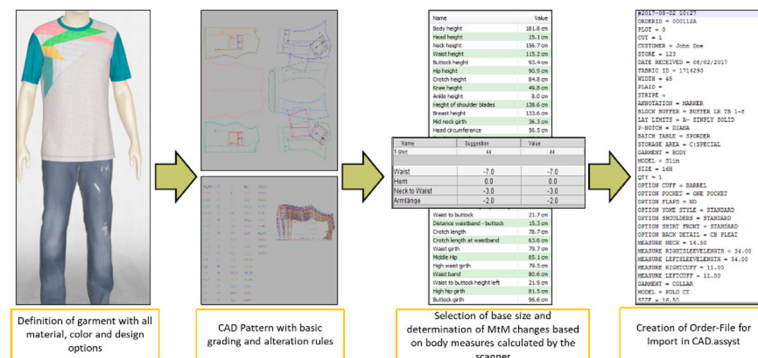


Fig. 2: Workflow component - from garment, pattern and body measurements to alteration values

2.2. From Body Measures to 2D Garment Pieces

The order file described in the previous section represents a basis for the automatic modification of the underlying MtM graded garment pattern according to the reference size and alteration values.

Which pattern measures can be altered depends on the pattern itself, its underlying grading rules and the body measures which were obtained from the 3D scan. In addition, the interdependence between these criteria must be considered. Typical circumferences and lengths for a shirt are e.g. neck, chest, waist and upper arm circumference as well as hem, sleeve length or back length. Figure 3 shows an example of an order file belonging to an MtM graded pattern of a women's T-shirt. It consists of basic customer information and the ordered quantity, as well as the information about pattern modifications. These in turn comprise the pattern name, reference size based on body scan as well as the altered measures and the alteration values.

```

1 CUSTOMER: Erika Mustermann
2 STYLES: MTM-WTSH
3 SIZE: 36
4 QUANTITY: 1
5 MEASUREMENTS: WAIST=-3.5,HEM=5.8,BACKLENGTH=0.0,NETOWAIST=0.0,UAW=3.2
    
```

Fig. 3: Example of an order file for a women's T-Shirt

The first step involves the import and processing of the order file by the MtM tool of the Assyst CAD Software Suite. Specifically, the software loads the MtM graded pattern including the predefined MtM rules for the basic size into the CAD program and generates a value table with alteration values from the order file. Subsequently, the alteration values from the value table are added to the basic size MtM final pattern measures. This results in a garment pattern in an individually modified reference size. Figure 4 shows the variation of a pattern during the described MtM workflow.

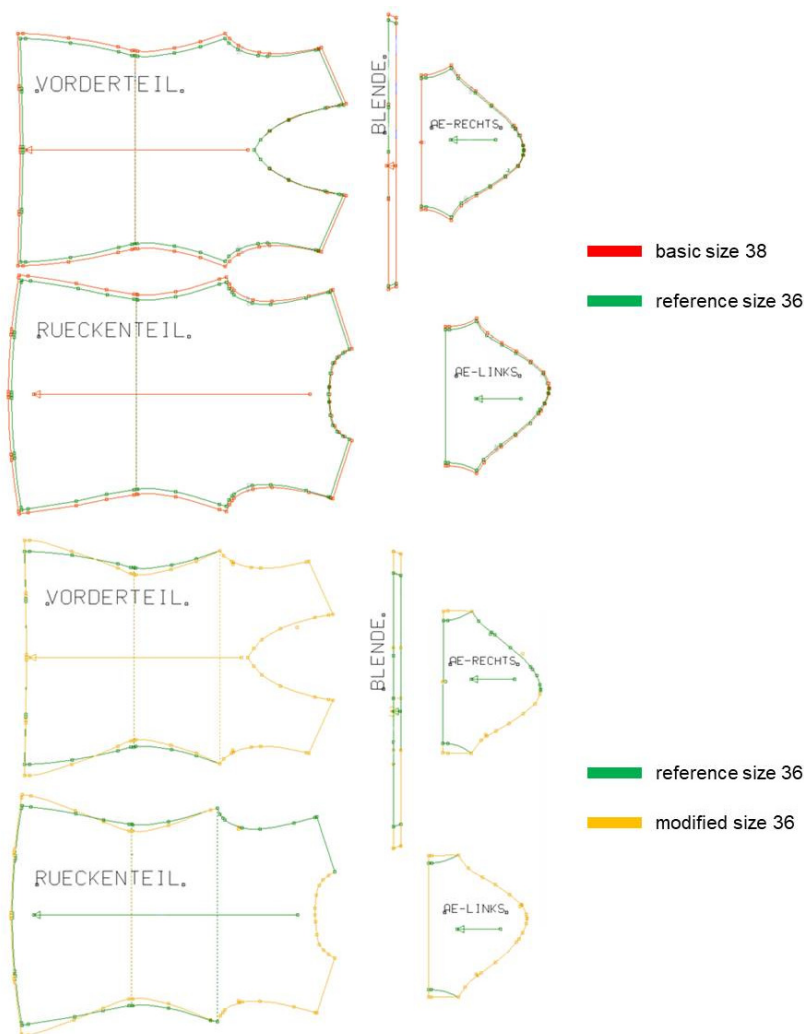


Fig. 4: Garment changes from basic size over reference size to individually modified size using order file values

In the final step, the data export tool of the Assyst CAD Software Suite is used to convert and save the pattern of the individually modified garment piece into the standardized ASTM file format for apparel [10]. This file format is a dialect of the DXF data format and simplifies the transfer of garment design data. The main advantage of the ASTM format is that different construction elements and metadata of the garment pattern, such as piece boundary contour and sew lines, are saved in separate and clearly defined layers. It is therefore easier to extract the relevant features of the garment piece for the subsequent knitting pattern generation.

2.3. Knitting Interpreter

This section describes the second main step of the scan-to-knit process, namely the automatic algorithmic generation of Jacquard knitting patterns from garment piece files. The algorithm is implemented in Python. Firstly, the procedure imports a garment DXF file in ASTM format. Each garment piece consists of a closed boundary which is discretized according to stitch parameters. The discretized geometry is further manipulated in 2D in order to optimize the knitting process. A tracing subroutine ensures the knitability of the piece, which is finally converted into a set of ASCII characters which can be imported and interpreted by Stoll knitting machine software.

In the following, we first list the input parameters for the algorithm and describe the file import and discretization routine. Subsequently, we provide an explanation of the 2D forming and tracing algorithm as well as the interface with the knitting machine software.

2.3.1. Input Parameters

For a proper automatic generation of knitting pattern for MtM garments, a garment piece file with clear and consistent structure is needed. The ASTM file format described in the previous section fulfills these requirements. Some of the most relevant ASTM layers for our algorithm are listed in table 1.

Table 1: Selection of layer definitions for ASTM format (based on [10])

| Layer | Definition |
|-------|----------------|
| 1 | Piece boundary |
| 7 | Grain line |
| 14 | Sew line |

In our typical use cases, each DXF file in ASTM format is divided into blocks. Each block contains a reference to a layer which is described by sorted lists of points and lines. Depending on the final assembly of the garment pieces, piece boundaries (layer 1) or sew lines (layer 14) can be used as piece boundary contours; in the former case, garment pieces are sewn together without seam allowance whereas in the latter case seam allowance is considered. Which boundary serves as the boundary contour for the final knitting pattern can be changed by assigning a Boolean input parameter to the algorithm.

The next important information in a DXF file in ASTM format regards the orientation of the pieces. In all of our use cases, the optimal piece orientation for flat knitting is the one opposite of the grain line in layer 7. Therefore, knitting (wales) directions are chosen towards the neck and shoulders for shirt pieces and towards waist for pants pieces.

Material and knit construction are taken into account by assigning the stitch dimensions for the algorithm, i.e. by providing a tuple of stitch width and stitch height. In the context of our algorithm, this parameter represents a sampling rate for the discretization of the garment piece contour. Stitch dimensions can be considered a virtual quantity, as they are usually given as number of stitches per length unit.

2.3.2. DXF File Import

The import of the garment DXF files is done using the Python Module `ezdxf` [11]. Most of the ASTM format layers are ignored as they are irrelevant for the Jacquard pattern. We thus only import the piece boundaries – either layer 1 or layer 14 – and the grain line layer 7 (see also Table 1). According to the layer 7 line direction the lines in the piece boundaries are rotated. In most of our cases, rotation for 90 degrees anticlockwise was necessary. Furthermore, as the base for the subsequent 2D manipulation we compute the feature points of the boundary, i.e. points where the angle between two neighboring lines of the piece contours is larger than a predefined tolerance. For our algorithm, this tolerance is set to 30 degrees. An example of a T-shirt with imported boundary contours and feature points is shown in figure 5.

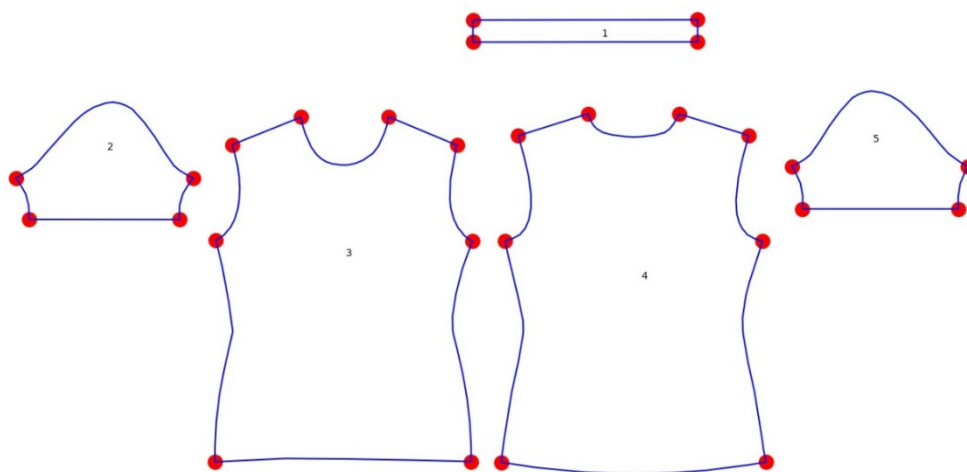


Fig. 5: An example of T-shirt pieces containing piece boundary contours and boundary feature points (red markers)

2.3.3. Piece Discretization

For each piece in a garment DXF file, a bounding box of the boundary contour is defined and overlaid with a rectangular mesh grid sampled according to stitch parameters. Thus, the points of the mesh grid represent stitches in the knitting machine program, although knitability is not yet guaranteed at this stage. To simplify the tracing algorithm, the double stitch height is taken instead of the original one. In our experiments, stitch parameters were sufficiently small so that the double stitch height did not visibly affect the accuracy of the discretization. From the stitch mesh grid, only points within the piece boundary are extracted. These points are subsequently interpreted as nodes in a graph structure with connecting horizontal and vertical edges between them (see figure 6). The rows in the knitting graph represent courses and the columns represent wales. The functionalities of the Python graph module `networkx` [12] are used in this step.

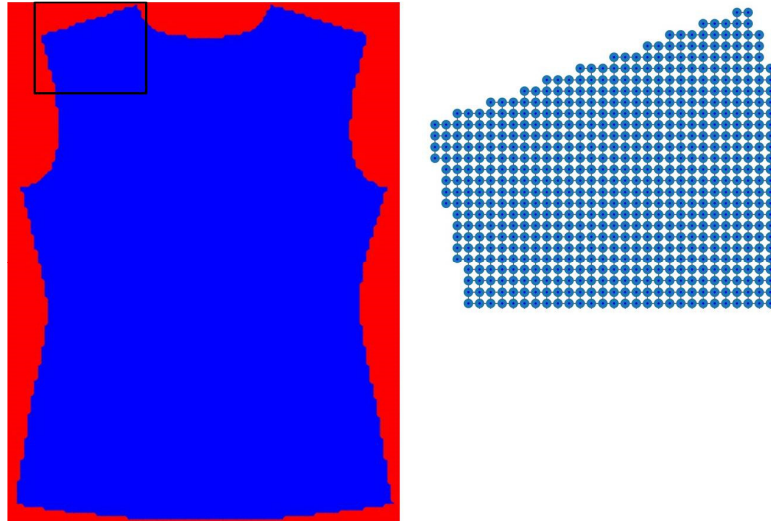


Fig. 6: A discretized shirt piece with the enclosing bounding box and a detail of the knitting graph for the left shoulder area (stitches are not to scale)

The boundary feature points are also discretized; a node attribute denoting a boundary feature is assigned to the nearest node in the graph which has less than four neighbors (i.e. it is the part of the discretized piece boundary).

2.3.4. 2D Forming

To ensure a better take-down behavior and smooth lower and upper edges of the knit pieces, the discretized graph needs to be altered accordingly. Specifically, goring is applied at the beginning and the end of the knit piece by extending the first and last horizontal row as shown in figure 7. The final width of such a row depends on the position of the closest boundary feature point.

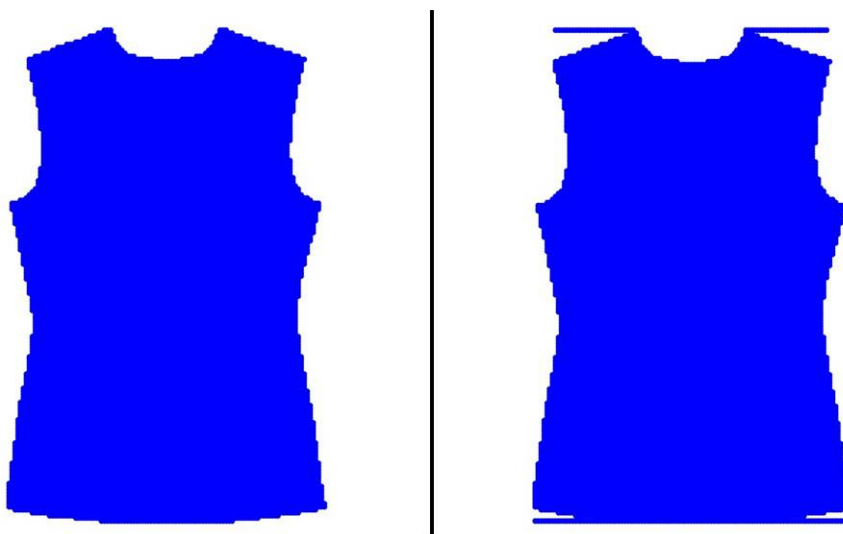


Fig. 7: A shirt piece before and after 2D forming. Note the addition of rows at the bottom and at the top.

However, in case of shirt torso pieces, goring in the last row is not applied over the entire width, as the additional stretch would cause a significant collar area distortion. This is particularly important for deep collars. This issue is solved by applying goring to both shoulder areas separately, extending two short shoulder-top rows outwards (as in figure 7). The consequence of this measure is that shoulder-tops cannot be knit with a single yarn, meaning that two yarn carriers must be used. After 2D manipulation, the pattern graph needs to be corrected for double stitch height. Every row is therefore doubled; additional nodes with horizontal edges between them and vertical edges are added and corrected accordingly.

2.3.5. Tracing and conversion to ASCII characters

The tracing algorithm alters the knitting graph so that it can be converted into a machine program. It is based on concept described by Narayanan et al. [8] and Liu et al. [6]. For our experiments, we determined a set of knitting rules which can be applied for several stitch constructions. These rules are fulfilled by the tracing algorithm and are regarding decreases and increases of the piece width as well as casting-off depending on the number of yarn carriers and their movement direction. The tracing algorithm alters the knitting graph according to these rules and assigns an appropriate attribute to every node representing a specific knitting operation. The rules are as follows:

1. The first yarn carrier starts at the right-hand-side of the needle bed and starts knitting to the left.
2. Second yarn carrier can only be used at the end of the knit piece.
3. If used, both yarn carriers move in the same direction.
4. No casting-on is allowed.
5. Decreases by one stitch are allowed in every row which does not belong to a gore.
6. Increases and casting-off stitches are allowed in the even rows on the left-hand-side boundary of the knit piece and in the odd rows on the right-hand-side boundary.
7. Increases should be no more than two stitches, preferably one stitch wide; larger increases would apply too much stress on the yarn.
8. A gore facing the left-hand-side boundary of the knit piece begins in an odd row and has an even number of rows.
9. A gore facing the right-hand-side boundary of the knit piece begins in an even row and has an even number of rows.
10. Both yarn carriers must finish on the left-hand-side.

It follows from the rules 1, 6, 8 and 9 and the doubling of the rows in the 2D forming step that knitability is almost entirely guaranteed for the left-hand-side boundary. Rule 10 is enforced by copying the last row. Only rule 7 is not enforced at this stage. However, most of our test data does not need large increases so that this extension of the tracing algorithm has been left for future work.

On the other hand, the right-hand-side boundary must be changed as the boundary changes contradict the knitting rules. The knitability on this boundary is therefore simply ensured by iterating through all the rows except for the last one and adjusting the right edge of the next row to match it with the edge of the current row (see figure 8).

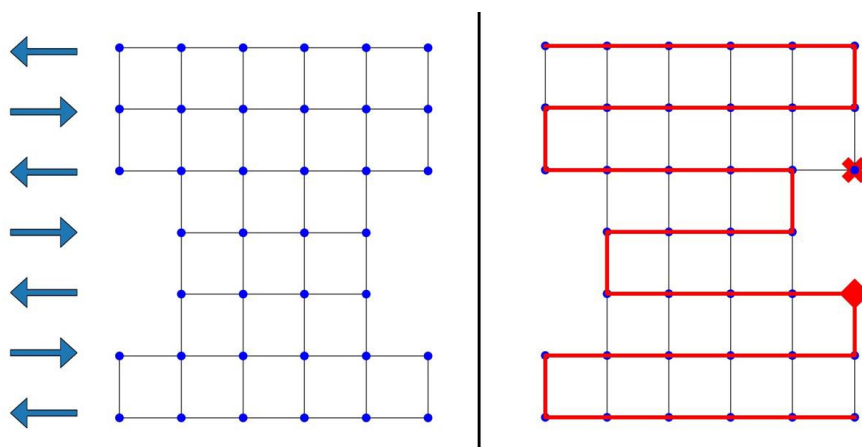


Fig. 8: An example of a knitting graph; arrows indicate the movement direction of the yarn carrier. Left: knitting graph before tracing. Right: knitting graph after tracing. The node marked with diamond is added to the graph while the node marked with X is removed from the graph. The thicker red line shows the movement of the yarn carrier during knitting.

Note that these manipulations provide a knitting graph which applies a more conservative rule for decreases than rule 5, since decreases only occur in every second row. This also means that many width decreases wider than one stitch have to be interpreted as casting-off. As this proved to pose no problems for the knitting process or the knit piece appearance, we leave this adjustment to future work. The result of the tracing algorithm is a graph which fulfills the knitting rules and therefore represents a knittable pattern. In the last step, every node of the graph needs to be converted to a proper ASCII character according to its attribute. To this end, we define a set of ASCII characters, each representing a normal stitch or a knitting operation as well as a not-in-use needle and held stitches in gore areas. The knitting operations are defined for both yarn carriers. Within Stoll's M1plus pattern software, each ASCII sign is assigned a color, as shown in the example in figure 9.

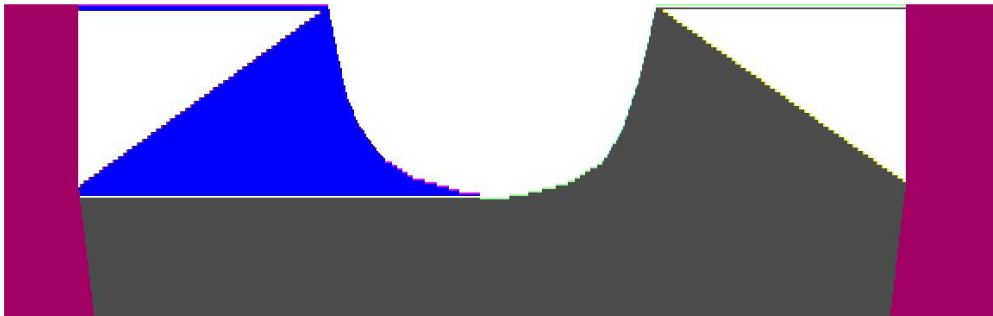


Fig. 9: A representation of a color-coded knitting pattern for a shoulder part of a shirt piece in Stoll's M1plus. Shoulders are knit with two different yarn carriers, hence different stitch colors. White areas are those where stitches are held on the needles.

3. Test/Data

Our method has been tested on different garment pieces and knit with several stitch constructions with different haptic and stretch properties and manufactured with three different yarn types. We have chosen four stitch constructions, the basic stockinette and garter stitches as well as two combinations of the basic stitches. The advantage of the chosen stitch constructions is that they provide sufficient variability for the experiments while only requiring a single knitting machine instruction set and consequently a single set of ASCII characters for all resulting knitting patterns.

The knitting instructions have been programmed for Stoll M1plus software and adapted for a Stoll ADF V-bed flat knitting machine with 899 needles and E18 gauge.

The development of a stitch simulation to predict the stitch dimensions for a particular stitch construction a priori was beyond the scope of our scan-to-knit project. Therefore, we needed to rely on experiments to determine the stitch parameters to be fed into the knitting interpreter. To this end, a test swatch as shown in figure 10 has been constructed and knit with the defined stitch constructions using realistic but arbitrary stitch parameters. After washing and drying, the swatches have been measured and compared to the measures of the CAD swatch model. This way, stitch parameters depending on yarn and stitch construction could be computed without much effort or material consumption.



Fig. 10: Test swatches to determine stitch parameters. Swatches based on the same geometry, knit with the same arbitrary stitch parameters, but with different stitch constructions, differ significantly.

After obtaining stitch dimensions for all the used stitch constructions, several different garment pieces were chosen as test data which were individualized for 3D body scans of volunteers.

4. Results

Based on 3D body scans of volunteers, the chosen MtM garment pieces have been altered and the corresponding order files generated. The resulting garment files in ASTM format were transformed into knitting programs using the knitting interpreter. The pieces were then manufactured, washed, dried and assembled.

As means of comparison and dimension validation, pieces were also assembled using the cut & sew process.

We have chosen two significant measures in horizontal and two in vertical direction for the validation. In case of a T-shirt, we measured the distance between the armpits and shoulder width as well as total height and sleeve height.

All the measurements exhibited less than 3% error when compared with cut and sewn garment pieces. An example of the measurements for a T-shirt is shown in table 2.

Table 2: Length comparison between a cut and sewn T-shirt and a knit and sewn T-shirt; all measures are in [mm].

| | | Measure | | | |
|-------------------|---------------------|--------------|---------------|--------------|-----------------|
| | | Total height | Sleeve height | Width armpit | Width shoulders |
| Knit construction | Length cut & sew | 675 | 208 | 511 | 461 |
| | Length knit | 677 | 207 | 520 | 467 |
| 1 | Absolute difference | -2 | 1 | -9 | -6 |
| | Relative difference | -0.29 % | 0.48 % | -1.76 % | -1.31 % |
| | Length knit | 665 | 205 | 517 | 466 |
| 2 | Absolute difference | 10 | 3 | -6 | -5 |
| | Relative difference | 1.48 % | 1.44 % | -1.17 % | -1.08 % |
| | Length knit | 665 | 205 | 522 | 469 |
| 3 | Absolute difference | 10 | 3 | -11 | -8 |
| | Relative difference | 1.48 % | 1.44 % | -2.15 % | -1.74 % |
| | Length knit | 665 | 209 | 519 | 468 |
| 4 | Absolute difference | 10 | -1 | -8 | -7 |
| | Relative difference | 1.48 % | -0.48 % | -1.57 % | -1.51 % |

We have thus demonstrated that all components of the scan-to-knit workflow work together seamlessly. This is possible on the one hand through the automation of the individual components involved, and on the other hand through the standardization of the interfaces in the data transfer between the modules. In the case of scanning and measurement extraction, this is achieved by a generalized approach, independent of the final product and is therefore, more general than the workflow afterwards. All scanning options used here result in the same type of inherent mesh, giving the option for standardized, semantic and topological information for the extraction of body dimensions.

For the application to made-to-measure garments an adaptation to the used examples is required. The garment (consisting of its final measurements, base size, grading, size recommendation, and alteration rules) has to be prepared and tuned. This step has to be done in advance, even before a scan-to-knit workflow can be started, as the workflow itself requires this kind of information for consistency. The garment depends on a couple of body measurements. Together with the scan of a customer, the result is an individual order file with alteration values. Subsequently, the process continues to run automatically, as long as the desired stitch construction is defined within the machine software and the knitting interpreter and its dimensions computed correctly. Up to the point where the Jacquard pattern is available, the whole workflow remains digital, making it quick, transparent and robust.

5. Conclusions

The focus of this work is to provide a technical solution to manufacture individualized knitwear. To this end, we developed a complete workflow concept starting with 3D scan and resulting in individually adapted garment pattern based on selected garment pieces. We implemented and integrated software modules for body segmentation, extraction of body measures, size recommendation and MtM processing. Furthermore, we developed an algorithm for automatic knitting pattern generation which considers material properties for a range of materials and stitch constructions. We successfully tested the algorithm for selected garment pieces. Within our institutions, we are able to demonstrate the integrated workflow for a selection of garment pieces and materials.

While the dimensions of the final pieces are accurate, we currently cannot validate the fit quality for the customer. As part of the future work, it is planned to measure 3D deformation of the garment when being worn. These results can then be compared with the a priori contact pressure simulation in Vidya as a validation method for the fit quality.

The overall workflow requires few user interventions. Currently unavoidable is however the expert knowledge when setting up the knitting machine as well as the manufacturing and assembly. Considering also washing and drying, these are the most time consuming workflow steps.

Nevertheless, the apparel industry can profit from our workflow since it enables fast and flexible individualized digitalized manufacturing of knitwear.

Acknowledgement

This work is funded by the Federal Ministry of Education and Research (BMBF, Germany) in the project Zwanzig20 – futureTEX –VP 26: Scan2Knit (project numbers 03ZZ0626A and 03ZZ0626B).

References

- [1] Shima Seiki WHOLEGARMENT, www.shimaseiki.com/wholegarment/, accessed 2020.
- [2] Stoll knit and wear, www.stoll.com/en/machines/knitwear/, accessed 2020.
- [3] Y. Igarashi, T. Igarashi and H. Suzuki, “Knitting a 3D Model”, in *Pacific Graphics*, Vol. 27, No. 7, 2008. <https://doi.org/10.1111/j.1467-8659.2008.01318.x>.
- [4] Y. Igarashi, T. Igarashi and H. Suzuki, “Knitty: 3d Modeling of Knitted Animals with a Production Assistant Interface”, in *Eurographics 2008 Annex to the Conference Proceedings*, 2008, pp. 187–190.
- [5] M. Popescu, M. Rippmann, T. Van Mele and P. Block, “Automated Generation of Knit Patterns for Non-developable Surfaces,” in *Humanizing Digital Reality*, Singapore, Springer Nature Singapore Pte Ltd., 2018, https://doi.org/10.1007/978-981-10-6611-5_56.
- [6] Y. Liu, L. Li. and P. F. Yuan, “A Computational Approach for Knitting 3D Composites Preforms”, in *Proceedings of the 2019 DigitalFUTURES*, Springer Nature Singapore Pte Ltd., 2020, https://doi.org/10.1007/978-981-13-8153-9_21.
- [7] J. McCann, L. Albaugh, V. Narayanan, A. Grow, W. Matusik, J. Mankoff, and J. Hodgins, “A Compiler for 3D Machine Knitting” in *ACM Transactions on Graphics*, Vol. 35, No. 4, 2016, <https://doi.org/10.1145/2897824.2925940>.
- [8] V. Narayanan, L. Albaugh, J. Hodgins, S. Coros and J. McCann, “Automatic Machine Knitting of 3D Meshes,” in *ACM Transactions on Graphics*, Vol. 1, No. 1, 2018, <https://doi.org/10.1145/3186265>.
- [9] V. Narayanan, K. Wu, C. Yuksel, and J. McCann, “Visual knitting machine programming”, in *ACM Transactions on Graphics*, Vol. 38, No. 4, Art. 63, 2019, <https://doi.org/10.1145/3306346.3322995>.
- [10] ePattern ASTM Standard dorthehansen.com/wp-content/uploads/2014/10/ePattern-ASTM-Standard.pdf, accessed 2020.
- [11] ezdx, ezdx.mozman.at/, accessed 2020.
- [12] A. A. Hagberg, D. A. Schult and P. J. Swart, “Exploring network structure, dynamics, and function using NetworkX”, in *Proceedings of the 7th Python in Science Conference (SciPy2008)*, pp. 11–15, 2008.