

# Wallpaper Pattern Alignment along Garment Seams

KATJA WOLFF and OLGA SORKINE-HORNUNG, ETH Zurich, Switzerland

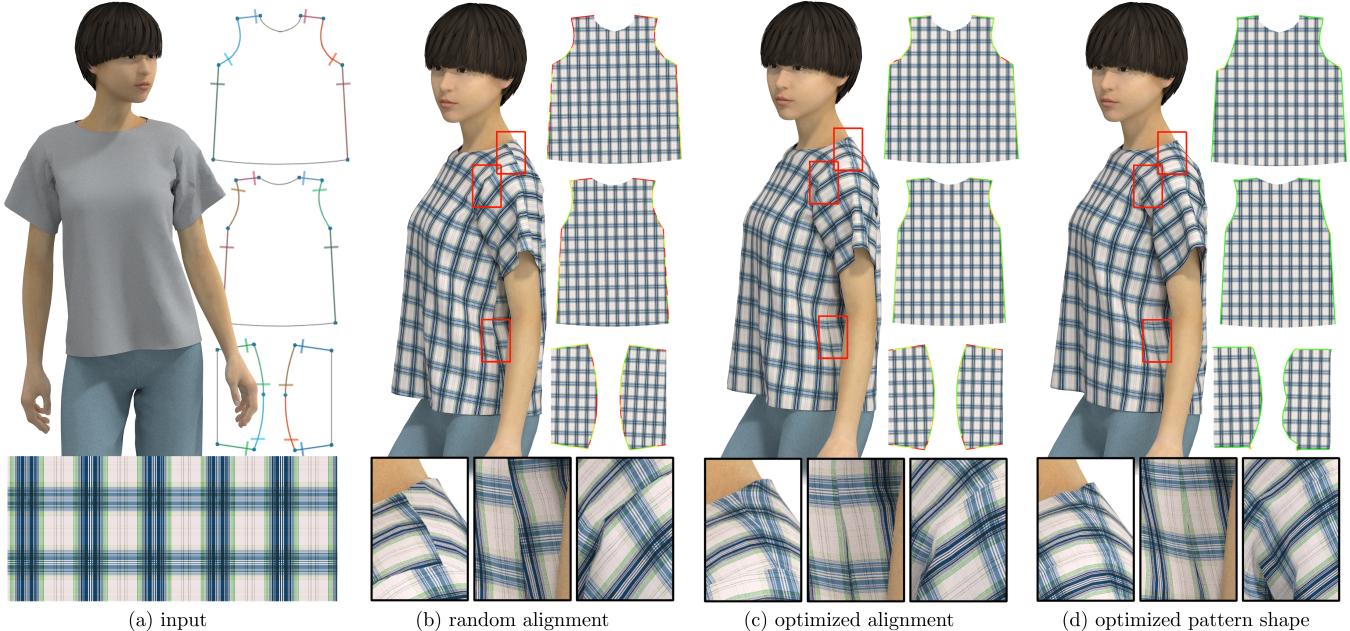


Fig. 1. Our method supports the production of high quality textured garment designs by optimizing the continuity and symmetry properties of the texture across seams. Our input is a sewing pattern and the desired fabric texture, or a 2D wallpaper pattern (a). A random placement of the sewing pattern on the fabric is likely to result in obvious misalignment of the texture on the seams, which is visually displeasing (b). Our method computes an optimal placement, such that misalignments of the repetitive wallpaper pattern are minimized (c). For most garment shapes, it is impossible to achieve a perfectly seamless appearance without changing the shape of the sewing pattern itself. Therefore we additionally provide the option to slightly alter the sewing pattern shape to obtain a much better texture fit along the seams (d). The improved visual appearance is noticeable in the 3D simulation of the garment, while the shape remains close to the original design. See also Fig. 3. The mismatch at each seam edge is visualized on a color scale from green (0 cm) to red (2 cm).

Despite recent developments towards on-demand, individualized garment design and fabrication, the majority of processes in the fashion industry are still inefficient and heavily dependent on manual work. A significant amount of recent research in this area has been focused on supporting designers to digitally create sewing patterns and shapes, but there is little work on textured fabrics. Aligning textile patterns like stripes or plaid along garment seams requires an experienced tailor and is thus reserved only for expensive, high-end garments. We present an interactive algorithm for automatically aligning repetitive textile patterns along seams for a given garment, allowing a user to make design choices at each step of our pipeline. Our approach is based on the 17 wallpaper groups and the symmetries they exhibit. We exploit these symmetries to optimize the alignment of the sewing pattern with the textured fabric for each of its pieces, determining where to cut

Authors' address: Katja Wolff, katja.wolff@inf.ethz.ch; Olga Sorkine-Hornung, sorkine@inf.ethz.ch, ETH Zurich, Switzerland.

Permission to make digital or hard copies of part or all 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 third-party components of this work must be honored. For all other uses, contact the owner/author(s).

© 2019 Copyright held by the owner/author(s).  
0730-0301/2019/7-ART62  
<https://doi.org/10.1145/3306346.3322991>

the fabric. We optionally alter the sewing pattern slightly for a perfect fit along seams, without visibly changing the 3D shape of the garment. The pieces can then be cut automatically by a CNC or laser cutter. Our approach fits within the pipeline of digital garment design, eliminating the difficult, manual step of aligning and cutting the garment pieces by hand.

CCS Concepts: • Computing methodologies → Computer graphics; Shape modeling; Mesh geometry models; Texturing.

Additional Key Words and Phrases: computational fabrication, garment modeling, wallpaper patterns, N-RoSy fields, non-convex optimization

## ACM Reference Format:

Katja Wolff and Olga Sorkine-Hornung. 2019. Wallpaper Pattern Alignment along Garment Seams. *ACM Trans. Graph.* 38, 4, Article 62 (July 2019), 12 pages. <https://doi.org/10.1145/3306346.3322991>

## 1 INTRODUCTION

Growing interest in fabrication processes in the computer science and engineering community, coupled with the dropping costs of machinery and material, have stimulated a lot of research on computational fabrication. The possibility to quickly produce real world-objects from digital models has transformed many sectors like 3D printing, architecture, and various industrial processes. Though



Fig. 2. Photograph of an inexpensive off-the-shelf shirt (left) and an expensive designer skirt (right). Notice how the repetitive print of the shirt clearly does not align along seams at the shoulder and below the arm, giving the garment an unrefined look. In contrast, the vertical alignment of the Burberry pattern of the skirt fits perfectly, and only the horizontal alignment shows a slight mismatch.

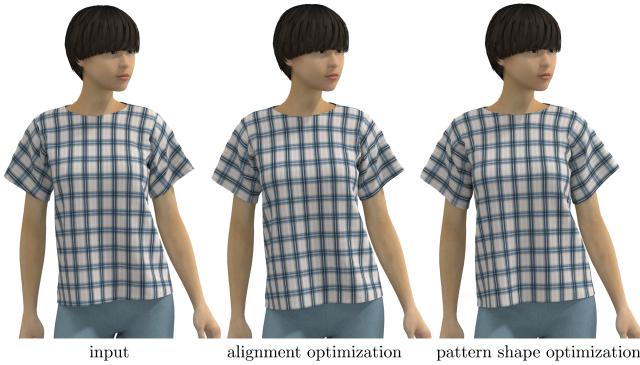


Fig. 3. Front view of the shirt from Fig. 1. Note how the garment shape created from the optimized sewing pattern (right) does not differ visibly from the input garment shape (left and middle).

there has also been a slow movement towards on-demand individualized fabrication in the garment industry fueled by recent research, many core fabrication techniques remain unchanged and still depend on strenuous manual labor.

Garments can be broadly categorized into two major groups by their fabrication process: Those created through knitting or crochet, and those made of flat panels of textiles sewn together. The garments in these two groups are inherently different in their structure, and we focus on the second type in this work. One aspect of such garment production process is the alignment of textured textiles along seams, which gives the garment a refined, high-quality look. To achieve a good fit, the garment's sewing pattern needs to be placed carefully on the chosen textile and cut such that the repetitive print aligns along the seams. This process involves an experienced tailor doing this work manually. Consequently, print patterns on mass-produced, affordable clothes often do not align along seams, as this time-consuming step is only profitable for high-end, expensive garments (see Fig. 2).

Many popular textile patterns in the garment industry, like the Burberry check, the houndstooth, polka dots or herringbone pattern (see Fig. 4) are repetitive and belong to the so-called wallpaper patterns. These can be categorized into 17 distinct groups with respect to occurring translational, rotational and reflectional symmetries (see Fig. 5). The human eye easily catches even the smallest misalignment in these patterns, and seam lines become obvious and negatively impact the overall impression of the garment (Fig. 1 (b) and Fig. 2). Garments with a fabric pattern that runs seamlessly or is reflected symmetrically across seams appear more elegant and refined.

We introduce a method to automatically compute the optimal alignment of a given wallpaper pattern to a given garment's sewing pattern in order to maximize the fit of the wallpaper along seams (Fig. 1 (c)). When we define fit in Sec. 3, we not only allow for a seamless continuation of the wallpaper across seams, but also take into account reflectional symmetries along seams, as they are also visually pleasing to the human eye. In contrast to works on mesh parameterization, which compute a mapping of a 3D model onto a set of flat charts, and some of which focus on seamless surface mappings (e.g. [Aiger et al. 2015]), in our setting the correspondence between the garment and the flat sewing pattern pieces, as well as the seams, are provided as input. Our core problem is instead precipitated by using a flat, rigid wallpaper and applying it to the 2D sewing pattern.

The first part of our approach is inspired by recent work on quad meshing, especially by Jakob et al. [2015]. Following their method, we separately optimize for rotational alignment while exploiting rotational symmetries of the wallpaper patterns (Sec. 4.1) and for translational alignment while taking into account repeating features of the pattern (Sec. 4.2). It is important to note here, that most 3D garment shapes do not allow for a perfect alignment along all seams, and we are often left with a very small, but nonetheless apparent discontinuity in the wallpaper pattern. Therefore, we additionally slightly adjust the sewing pattern if necessary (Fig. 1 (d)). This greatly improves the perceived fit along seams, but at the same time, since the changes to the sewing pattern are small, the difference in the 3D garment shape is negligible and hardly perceivable (see Fig. 1 and Fig. 3).

Nowadays, a multitude of commercial software supports the digital creation of garments (e.g. [CLO 2018]), including the design of sewing patterns and the simulation of the finished garment on a virtual model, allowing for different choices of materials. Making design changes digitally removes the need for repetitive fitting on a real model in the traditional apparel design process. A finished sewing pattern can be exported and used just as a traditionally designed one. Our method fits neatly between these two stages and is highly interactive, with each step being independent of the others, optional and allowing for user input and adjustments for individual seams or sewing pattern pieces. Given the computed alignment for each sewing pattern piece, it becomes possible to cut these pieces automatically out of the fabric using a laser or CNC cutter. This way, the textile pattern alignment becomes part of the design process, and complicated manual alignment and cutting can be eliminated.



Fig. 4. A selection of photographed wallpaper patterns on textiles and their corresponding wallpaper groups (photos taken by us). Included are common textile patterns like the houndstooth pattern (second row, first), the herringbone pattern (second row, third), the polka dots (second row, last) and the Burberry check (third row, first).

## 2 RELATED WORK

We review the most related literature on virtual garment design, as well as works concerned with symmetry and wallpaper groups.

**Garment design.** Related work that addresses the creation and alteration of sewing patterns, as well as interactive garment design, all focuses on the shape of the garment. One work by Lu et al. [2017] proposes a method to design a print directly on the 3D model and transfer it to the 2D sewing pattern, but to the best of our knowledge, our work is the first to incorporate the alignment of periodic textures in the garment design process.

The survey by Nayak et al. [2017] gives a broad overview of the use of automation in garment manufacturing, including computer aided design. Early works on garment design allow for interactive sewing and cutting of a garment during a physics based cloth simulation, as well as automatic adjustments of the underlying sewing pattern [Keckeisen et al. 2004]. Volino et al. [2005] extensively review contemporary techniques of that time for garment design and detail a framework that concentrates on interactive design, simulation and visualization features.

Sketch based methods [Decaudin et al. 2006; Robson et al. 2011; Rose et al. 2007; Turquin et al. 2007] enable the user to design a 3D pattern by drawing contours, borderlines and seamlines directly onto a virtual mannequin. They automatically compute developable patches for the sewing pattern, which can then be used to create a real garment by sewing the pattern parts together. Umetani et al. [2011] allow for bidirectional interactive garment design, in which both the 3D and the 2D pattern can be edited, and their correspondence is automatically maintained. To enhance the traditional workflow of pattern design, instead of working only digitally, a physical real-world mannequin can be used as a guide, combined with a tool for drawing in 3D around it to design garments [Wibowo et al. 2012]. Kwok et al. [2016] use an evolution method based on aesthetic factors of styling curves of a garment to automatically generate new

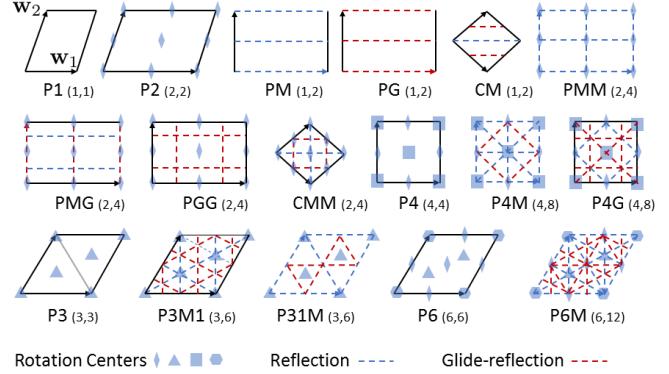


Fig. 5. The primitive cell for each of the 17 wallpaper groups and the occurring symmetries. The name of the wallpaper group along with its maximal rotational symmetry and the number of equivalent points in a cell is denoted below each diagram. The arrows symbolize the two translational vectors  $w_1$  and  $w_2$ .

garment designs. Recent work incorporates advancements in machine learning to develop a data-driven approach, which uses a sketch of desired fold patterns to estimate corresponding garment shapes via a novel shared shape-space [Wang et al. 2018].

Instead of creating new sewing patterns from scratch, [Berthouzoz et al. 2013] scan and parse existing, traditionally published patterns and convert them into 3D models. Other works adjust and combine existing digital sewing patterns. For example, [Brouet et al. 2012] grade existing patterns for different body shapes while preserving style and fit, and [Bartle et al. 2016] allow users to mix existing garment designs or adjust for length and fit via a physics-based draping simulation. Another sketch based approach by Li et al. [2018] enriches a garment with folds and pleats by computing both a new 2D pattern and a 3D garment guided by sketches. To facilitate real-world fabrication of computed patterns, Igarashi and colleagues [2008] describe how to generate a necessary seam allowance to sew 2D pieces cut from real fabric together. Similarly to these works, we adjust an existing sewing pattern in order to allow for a better fit of the texture along seams.

**Wallpaper groups and symmetry.** Wallpaper patterns and their categorization into the 17 wallpaper groups as first proven by Fedorov [1891] have long been established in mathematics. We exploit the symmetries exhibited by the 17 groups to align repetitive patterns on textiles along garment seams. Recent work in the computer vision community [Liu et al. 2004] describes how to use wallpaper group theory to automatically classify the symmetry of a given pattern and extract the representative motif that repeats in the pattern, which can be applied to textile prints as well. The thesis of von Gagern [2008] deals with the foundations of drawing symmetric ornaments in the 2D plane and explains the steps for the recognition of symmetry groups of patterns from external images, including available source code. A comprehensive review of relevant mathematical theory and computational treatment of symmetry and group theory in computer vision and computer graphics is given by Liu

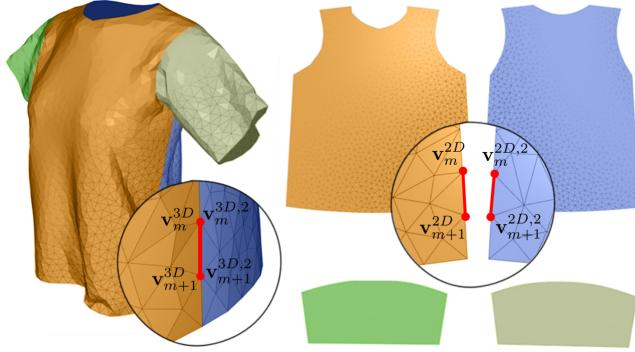


Fig. 6. A simulated 3D garment (left) and its corresponding sewing pattern (right). Both meshes consist of the same amount of vertices and exactly the same faces. The different  $N_p = 4$  sewing pattern pieces are shown with different colors. Border edges of two pattern pieces can coincide in the 3D garment and create a seam edge (red line). There are 8 seams connecting the pattern pieces of this shirt.

et al. [2010], including symmetry in the case of 2D wallpaper patterns. The state-of-the-art report by Mitra et al. [2013] surveys and classifies works in symmetry detection, focusing not only on 2D repetitive patterns, but also including works in the larger field of 3D shape understanding. Tyler [2003] reviews the human perception of the aesthetics of symmetry and repetitive patterns, which influences our definition for fit along seams in Sec. 3.

One type of symmetry present in some wallpaper groups is  $N$ -directional rotational symmetry ( $N$ -RoSy), which has found applications in many other areas of research in computer graphics, reviewed in the survey by Vaxman and colleagues [2016]. Our work is related to the use of  $N$ -RoSy fields for quad-dominant and triangle remeshing as proposed in [Jakob et al. 2015], though we work with a larger number of different  $N$ -RoSy's and additional reflectional symmetry (see Sec. 4.1).

### 3 METHOD OVERVIEW

Our input is a textile print pattern, which we call *wallpaper* for the rest of this paper and further define below, a 2D garment sewing pattern, which we call simply the *pattern*, and the corresponding simulated 3D *garment*, in which the different *pattern pieces* are connected at the seams (see Fig. 6). Our main objective is to create an optimal fit of the wallpaper along the seams. In this section, we define the needed terminology and formalize the meaning of fit. We also describe the two major parts of our pipeline, as shown in Fig. 1: The wallpaper alignment optimization step, in which we do not make any changes to the pattern shape but only its placement on the fabric, and the pattern optimization step, in which the wallpaper alignment is fixed and instead deformations are applied to the shape of the pattern pieces.

*Garment, sewing pattern and seams.* We denote the input 3D triangle mesh representing the simulated garment by  $M^{3D} = (V^{3D}, F^{3D})$  and the 2D mesh representing the 2D sewing pattern by  $M^{2D} = (V^{2D}, F^{2D})$ , where  $V^{3D}$  and  $V^{2D}$  are the sets of vertices and  $F^{3D}$  and  $F^{2D}$  the sets of faces. The 3D garment  $M^{3D}$  is a deformation

of the 2D pattern  $M^{2D}$ , therefore the number of vertices in both meshes is the same and they have the same connectivity  $F^{2D} = F^{3D}$ . The pattern mesh  $M^{2D}$  can be regarded as a parameterization of  $M^{3D}$ . Note that  $M^{3D}$  and  $M^{2D}$  are generally not locally isometric, because cloth tends to stretch due to gravity and placement on an avatar. Both meshes consist of the same number  $N_p$  of connected components, i.e., the individual pattern pieces. The connected components of the pattern are clearly visible and separate, while those of the 3D garment meet at the seams. We denote by  $\mathcal{S}$  the set of the given seams on the garment. A seam  $S$  between two pattern pieces  $P_i$  and  $P_j$  is a set of pairs of consecutive boundary edges from  $P_i$  and  $P_j$ , i.e.,  $S = \{(e_k, e'_k)\}$ . We say a seam of the 3D garment consists of twin *seam lines* in the 2D sewing pattern. Note that two pattern pieces can share more than one seam (e.g., the front and back of a shirt share two side seams and two shoulder seams), and a pattern piece can have a seam with itself (then  $i = j$ , e.g., a dart or the sleeve of a shirt). The garment and its pattern mesh can be created with existing garment modeling software, e.g. [CLO 2018]. At any point in the pipeline we allow the user to specify a weight  $w_S$  for each seam, to either ignore seams ( $w_S = 0$ ) in the optimization (for example those that are not visible when the garment is worn and therefore irrelevant for the wallpaper fit) or increase the importance of the fit for seams that are very prominent on the garment.

*Wallpaper groups.* We give a short introduction to the concept of wallpaper patterns, as far as is necessary for understanding the methods introduced in this paper. For an in-depth discussion of the wallpaper group theory and how wallpaper patterns fit into the more general topic of symmetries, refer to [Liu et al. 2010]. A wallpaper pattern is a periodic pattern in the 2D plane, including patterns like plaid or periodic dot patterns (see Fig. 4 for examples of such patterns in textiles). Due to the periodic nature of such patterns, they exhibit a range of symmetries, with the simplest being a plain translation present in any wallpaper. Each wallpaper pattern lattice is generated by two smallest translation vectors  $w_1$  and  $w_2$ , which divide the plane into identical parallelograms, called tiles or primitive translational cells. Additional symmetries are 2, 3, 4 and 6-fold rotational symmetries, as well as reflections and glide-reflections along an axis. Depending on the exhibited symmetries, all wallpaper patterns can be categorized into exactly 17 wallpaper groups, as first proven by [Fedorov 1891]. The primitive cell for each of the 17 groups and the corresponding symmetries are shown in Fig. 5.

We specify a given wallpaper pattern by an integer that identifies the wallpaper group, the two translational vectors  $w_1$  and  $w_2$ , an origin  $w_o$  of the basic translational cell in the 2D plane and an image file containing color values for the cell. Given a printed textile, we can either measure and determine these parameters by hand or use existing methods to automatically extract this information from the image [Liu et al. 2004; von Gagern 2008]. Figure 7 shows an overview of how we apply the texture to the 3D garment and 2D sewing pattern. For each component of the 2D sewing pattern we define a rigid transformation consisting of a 2D rotation matrix  $R_c$  and a 2D translation vector  $t_c$ , which maps points  $p^{2D}$  of the pattern into points of the the wallpaper space  $p^{WP}$ :  $p^{2D} = R_c p^{WP} + t_c$ . The basis transformation matrix  $W = [w_1 \ w_2]$  gives us the position

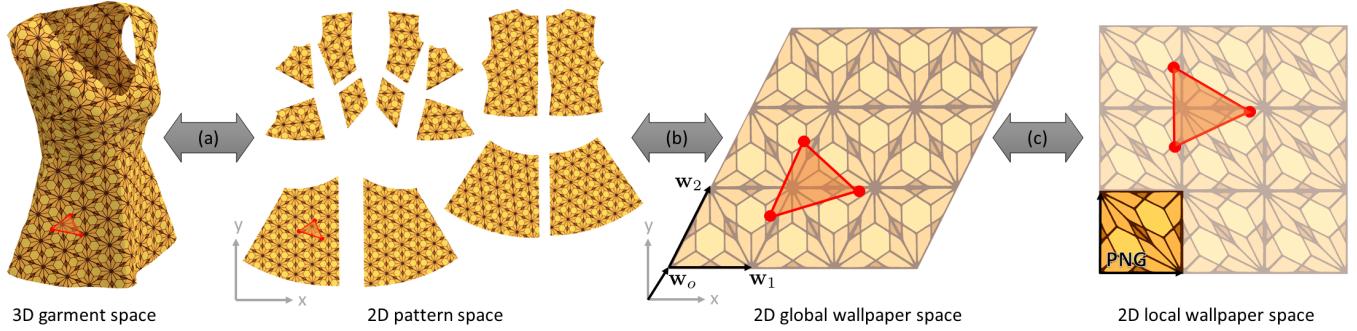


Fig. 7. Overview of the connection between the 3D garment space, the 2D sewing pattern space, the 2D wallpaper space and the 2D local wallpaper space. Since we know the correspondences between the vertices of the 3D garment and those of the 2D pattern, we can easily look up the corresponding 3D position of a 2D vertex and vice versa (a). For each component  $P_c$  of the sewing pattern we define a rotation  $R_c$  and translation  $t_c$  that map the vertices of the pattern into the wallpaper space (b). A basis transform maps the translational vectors  $w_1$  and  $w_2$  to standard unit vectors (c). This allows for easy mapping into the basic translational cell and the image file coordinates.

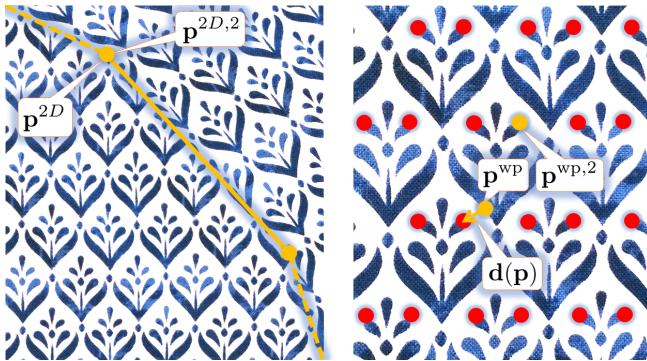


Fig. 8. Measuring the fit at a given point on a seam. On the left we illustrate a seam between two pattern pieces, with its seam edges and vertices; the texture is a wallpaper of the CM group. Each location on the seam in 3D belongs to two different pattern pieces and therefore has two corresponding locations in the wallpaper plane,  $p^{wp}$  and  $p^{wp,2}$ . We define the mismatch at a seam point as the distance between  $p^{wp}$  and the closest of all equivalent points of  $p^{wp,2}$  (in red).

of  $p^{wp}$  in the local wallpaper space:  $p^{\text{local}} = W^{-1}(p^{wp} - w_o)$ . The rounded integer coordinates of  $p^{\text{local}}$  describe the cell in which the vertex lies, and the fractional part of the coordinates gives us the location in the associated image file.

For each point  $p^{wp}$  in the wallpaper plane, we can define its *equivalent points* as all points that can be generated by applying the wallpaper symmetries to  $p^{wp}$ . Figure 8 (right) shows an example of equivalent points. Given a point in the wallpaper plane, we can create some of the equivalent points by translation by integer multiples of  $w_1$  and  $w_2$ , which we call equivalent points of the same type. Each wallpaper group creates a different number of equivalent points in one single cell, denoted in Fig. 5, which is equal to the number of different types of equivalent points.

**Fit along seams.** To define the quality of the fit along a seam, we need to understand what appeals to the human perception. Clearly, a seamless continuation of the wallpaper across a seam renders the

seam invisible and can therefore be considered a perfect fit. Additionally, humans have an innate tendency to recognize symmetry and patterns, due to their pervasiveness in nature and man-made structures. Symmetry has a perceptual appeal to the human brain, as witnessed throughout history by the universal use of symmetry in art, decor, architecture and the design of wallpaper patterns [Tyler 2003]. We therefore define the fit quality in a way that favors both seamless continuation of the wallpaper and reflective symmetry at seam edges.

We first define the fit measure at a single point  $p$  on a seam, as illustrated in Fig. 8. This point belongs to two different pattern pieces and therefore has two different 2D coordinates  $p^{2D}$  and  $p^{2D,2}$  corresponding to two different points in the wallpaper plane:  $p^{wp}$  and  $p^{wp,2}$ . Each of these points has infinitely many equivalent points. We define the fit at  $p$ , denoted by  $d(p)$ , as the smallest distance between equivalent points of  $p^{wp}$  and  $p^{wp,2}$ , or equivalently, the distance between  $p^{wp}$  and the closest equivalent point of  $p^{wp,2}$ . We denote the vector between this pair of closest equivalent points by  $d(p)$ , in the following always pointing away from the currently regarded pattern piece  $P_c$ . If the distance is zero, we have a perfect fit in this seam point. For a whole seam, we integrate the fit values of all seam points. In our discrete case, we simply sum over the fit at the midpoints of the seam's edges, weighted by the respective edge length. The overall fit measure for a garment is therefore

$$E_{\text{fit}} = \sum_{S \in \mathcal{S}} w_S \sum_{(e, e') \in S} l_e d(v_e), \quad (1)$$

where  $w_S$  is the weight defined for each seam,  $l_e$  is the edge length and  $v_e$  is the midpoint of seam edge  $e$ . The measure  $d(v_e)$  can vary for different edges belonging to the same seam, and we therefore need to regard each seam edge individually. If a wallpaper seamlessly continues across a seam, we obviously get a perfect fit value of zero, and the above definition also accommodates for the case of a perfect fit, where the wallpaper is perfectly reflected along a seam.

**Wallpaper alignment optimization (Sec. 4).** In this step, we optimize the alignment of the wallpaper for the  $N_p$  pattern pieces, which



Fig. 9. A simple example of a garment and a wallpaper that do not admit a perfect fit, due to the shape of the side seams. Along the visible seam on the garment worn by the model, we can see that the spacing between the stripes touching the seam varies from top to bottom. On the left side of the seam, the spacing becomes wider towards the top and on the right side of the seam it is the opposite, making it impossible to fit the wallpaper perfectly along this seam without changing the shape of the pattern.

means optimizing all rotations  $R_1..R_{N_p}$  and translations  $t_1..t_{N_p}$ . Inspired by approaches in quad meshing that also take into account rotational and translational symmetries, and especially by the work of Jakob et al. [2015], we further subdivide this step into first computing a rotational symmetry (RoSy) field on the pattern pieces, thereby determining  $R_1..R_{N_p}$ , and secondly a positional field, determining  $t_1..t_{N_p}$ . Both steps are independent of each other, and designers might for example choose to set the orientation of the wallpaper on their own and only run our optimization for the translation. Note that for most garments, a perfect fit is not attainable by optimizing the alignment alone (see Fig. 9).

*Sewing pattern optimization* (Sec. 5). In this step, we keep the alignment  $R_i, t_i$  of the wallpaper for each pattern piece  $P_i$  fixed and instead adjust the shape of the pattern pieces in order to achieve an optimal fit. We first compute an ideal shape for each seam in the 2D plane, independently. We then restore a coherent shape for the pattern pieces by deforming them via a global energy minimization. The optimized 2D pattern can then be re-imported into a garment simulation software to view the resulting shape in 3D. After applying the wallpaper alignment optimization in the previous step, the needed changes to the pattern are often small, and therefore the changes to the 3D shape are negligible, while eliminating the remaining mismatches of the wallpaper along the seams. Perceptually this makes a significant difference for the human eye.

#### 4 WALLPAPER ALIGNMENT OPTIMIZATION

In this section, we describe in detail how we optimize for a wallpaper alignment that minimizes the mismatch along seams while keeping the shape of the pattern fixed. We exploit occurring symmetries of the wallpaper, especially rotational symmetries (RoSy).

Our approach adapts the work by [Jakob et al. 2015], which leverages N-RoSy’s to create quad-dominant and triangle meshes. Our input representation differs starkly from a simple point cloud with neighbourhood information as in their method: we deal with pattern pieces connected via pairing of many individual seam edges. Nevertheless, we can adopt their approach of separately optimizing all rotations  $R_1 .. R_{N_p}$  and only afterwards optimizing the translations  $t_1 .. t_{N_p}$ . We apply their intrinsic measure for rotational fit and adapt their optimization algorithm to our problem, additionally incorporating a measure of symmetry along seams. The first step of optimizing rotations can be seen as a foundation for the second step, where we find  $t_1 .. t_{N_p}$  that optimize the fit measure defined in Eq. (1).

##### 4.1 Rotational alignment

To calculate the rotation of the wallpaper for each pattern piece, we need to take into account the rotational symmetries present in the wallpaper. Any wallpaper might exhibit a combination of 2, 3, 4 or 6-fold RoSy, and we exploit the highest  $r$ -RoSy (denoted in Fig. 5 for each wallpaper), giving us additional freedom in fitting the wallpaper along the seams. This means that the rotation for each pattern piece is unique up to the  $r$ -RoSy.

*Measuring rotational fit along seams.* Our goal is to create a smooth orientation field across seams, and our measure is inspired by the intrinsic smoothness measure in [Jakob et al. 2015]. We do not need the wallpaper to align with 3D features like folds and wrinkles that are serendipitously generated by the cloth simulator, so we do not incorporate the extrinsic measure proposed in [Jakob et al. 2015]. Nevertheless, the 3D simulated garment plays a vital role in our computations, since its shape incorporates the specific physical properties of the chosen material, realized by the cloth simulator. The intrinsic deformations of the triangles in the 3D shape influence the orientation of the wallpaper, and therefore we use the 3D garment as our reference shape instead of computing the wallpaper alignment solely based on the 2D sewing pattern.

Given a seam between two pattern pieces  $P_i$  and  $P_j$ , we consider each individual pair of twin seam edges,  $(e, e')$ . They have adjacent faces  $f_e$  and  $f'_e$  belonging to  $P_i$  and  $P_j$  respectively, with face normals  $\mathbf{n}_e$  and  $\mathbf{n}'_e$  in  $M^{3D}$  (see Fig. 10). The faces  $f_e$  and  $f'_e$  are also associated with orientation vectors  $\mathbf{o}_e$  and  $\mathbf{o}'_e$ , which represent the orientation of the wallpaper in 3D. To obtain these, we take an arbitrary 2D vector in the wallpaper space as a representative (e.g., simply  $\mathbf{w}_1$ ) and transform it for each pattern piece  $P_i$  and  $P_j$  into the 2D pattern space by applying  $R_i$  and  $R_j$ . We denote these representative rotation vectors by  $\mathbf{r}_i$  and  $\mathbf{r}_j$ . We then transfer this to the corresponding triangle in the 3D garment space as illustrated in Fig. 7. For simplicity, we denote the transformation matrix to calculate  $\mathbf{o}_e$  and  $\mathbf{o}'_e$  as  $F_e$  and  $F'_e$  respectively, with  $\mathbf{o}_e = F_e \mathbf{r}_i$ , and describe in more detail how to calculate  $F_e$  in the appendix. The vectors  $\mathbf{o}_e$  and  $\mathbf{o}'_e$  vary along each edge of the seam, making it necessary to calculate rotational fit for each edge separately.

For a single seam edge, we can measure the fit between both orientations intrinsically as the angle difference between the vectors  $\mathbf{o}_e$  and  $\mathbf{o}'_e$  after unfolding them onto a common plane and explicitly taking into account the ambiguity introduced by the  $r$ -RoSy, as done

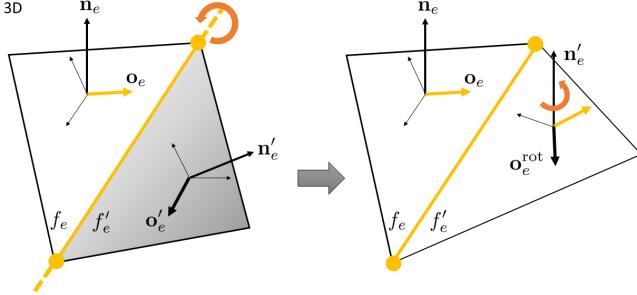


Fig. 10. Measuring the fit of a 3-RoSy wallpaper intrinsically at a seam edge (yellow). Neighboring faces  $f_e$  and  $f'_e$  have normals  $n_e$ ,  $n'_e$ , respectively (left). We measure orientation fit intrinsically, similar to [Jakob et al. 2015], by rotating the face  $f'_e$  by the angle  $\angle(n'_e, n_e)$  about axis  $n'_e \times n_e$ , such that it becomes coplanar with the face  $f_e$  (right). We can then measure the difference between the corresponding orientations  $\mathbf{o}_e$  and  $\mathbf{o}'_e$  with respect to the RoSy in the common 2D plane, by finding the integer rotation of  $\mathbf{o}'_e$  that has the smallest angle with  $\mathbf{o}_e$  (yellow arrows).

in [Jakob et al. 2015] (see Fig. 10). To take the rotational symmetries into account, integer rotations of a representative vector  $\mathbf{o}$  about a fixed normal  $\mathbf{n}$  are defined as

$$\mathcal{R}_r(\mathbf{o}, \mathbf{n}, k) := \text{rot}(\mathbf{n}, k \frac{2\pi}{r}) \mathbf{o}, \quad k \in \mathbb{Z}, \quad (2)$$

where  $\text{rot}(\mathbf{a}, \phi)$  is a rotation about axis  $\mathbf{a}$  by angle  $\phi$ . In contrast to [Jakob et al. 2015], we define the global energy measuring the rotational fit by summing over all seam edges, instead of vertices and their neighbors:

$$E_{\text{rot}} = \sum_{S \in \mathcal{S}} w_S \sum_{(e, e') \in S} l_e \angle(\mathbf{o}_e, \mathcal{R}_r(\mathbf{o}_e^{\text{rot}}, \mathbf{n}_e, k_e^{\text{rot}}))^2, \quad (3)$$

where  $\mathbf{o}_e^{\text{rot}} := \text{rot}(\mathbf{n}_e \times \mathbf{n}_e, \angle(\mathbf{n}_e, \mathbf{n}_e)) \mathbf{o}_e'$  is the representative vector  $\mathbf{o}_e'$  for face  $f'_e$  rotated into the plane of face  $f_e$ , and  $k_e^{\text{rot}}$  is an integer variable defined for each seam edge. Note, that we can ignore seam edges connecting a pattern piece to itself, since the rotation is the same on both sides and so any rotation would result in the same fit value; omitting such seam edges reduces computation time.

*Rotational energy minimization.* We minimize the energy (3) as described in [Jakob et al. 2015]. In our case this amounts to iteratively updating the representative rotational vectors  $\mathbf{r}_c$  of all pattern pieces  $P_c$ . In one iteration, we go through all seam edges of  $P_c$  and update  $\mathbf{r}_c$  after visiting each edge. Since the orientation vectors of all edges are based on the rotation vectors  $\mathbf{r}_c$ , we first calculate the orientation vectors  $\mathbf{o}_e = F_e \mathbf{r}_c$  and  $\mathbf{o}'_e = F'_e \mathbf{r}_c'$  as described in the appendix for the current edge. The update is then done as follows:

$$\mathbf{o}_e \leftarrow \mathcal{R}_r(\mathbf{o}_e^{\text{rot}}, \mathbf{n}_e, k_e^{\text{rot}}), \quad (4)$$

and with this updated orientation vector, we in turn update the pattern piece's rotation vector  $\mathbf{r}_c$ , taking into account the seam weight  $w_S$  and edge length  $l_e$ :

$$\begin{aligned} \mathbf{r}_c &\leftarrow \mathbf{r}_c + w_S l_e F_e^{-1} \mathbf{o}_e \\ \mathbf{r}_c &\leftarrow \mathbf{r}_c / \|\mathbf{r}_c\|. \end{aligned} \quad (5)$$

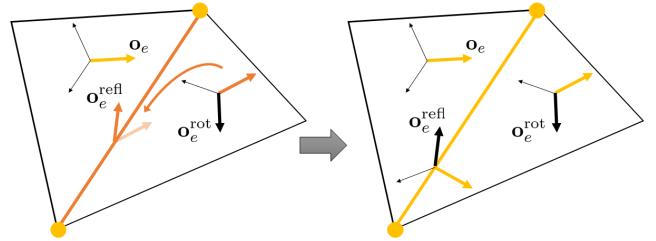


Fig. 11. Taking reflectional symmetry along the seam edge into account. We reflect the closest integer rotation of the vector  $\mathbf{o}_e^{\text{rot}}$  along the seam edge and denote this vector by  $\mathbf{o}_e^{\text{refl}}$  (left, orange vectors). Again taking the RoSy into account, we find the integer rotation of  $\mathbf{o}_e^{\text{rot}}$  that has the smallest angle with  $\mathbf{o}_e$  (right, yellow vectors).

The vector  $\mathbf{o}_e^{\text{rot}}$  is computed as described above, and we locally compute the optimal variable  $k_e^{\text{rot}}$  each time by a small brute-force search:

$$\mathbf{k}_e^{\text{rot}} := \underset{0 \leq k < r}{\operatorname{argmin}} \angle(\mathbf{o}_e, \mathcal{R}_r(\mathbf{o}_e^{\text{rot}}, \mathbf{n}_e, k)). \quad (6)$$

We iterate over all pattern pieces until the total change between the rotation vectors  $\mathbf{r}_c$  and their latest update  $\mathbf{r}'_c$  becomes negligible, i.e.,  $\sum_{c=1..N_p} \|\mathbf{r}_c - \mathbf{r}'_c\|^2 < \epsilon$  with  $\epsilon = 10^{-7}$ . We illustrate in Fig. 12 (b) the results of the above algorithm on a simple example. It is apparent that our approach minimizes angle differences, but at the same time might lead to rotations along seams that do not allow for a perfect translational fit. Therefore we expand this approach by allowing for symmetries along seams in the next paragraph.

*Seam reflection.* As discussed in Sec. 3, we do not only strive for a perfect, seamless fit along seams (which is infeasible for almost all garments), but also include reflectional symmetry *along seams*, increasing the degrees of freedom when aligning the wallpaper. Note that this type of symmetry differs from the symmetries exhibited by the wallpaper groups. In terms of the algorithm, this means that we not only regard the difference between the orientation vectors  $\mathbf{o}_e$  and  $\mathbf{o}'_e$ , but also the difference between  $\mathbf{o}_e$  and the reflection  $\mathbf{o}_e^{\text{refl}}$  of  $\mathbf{o}_e^{\text{rot}}$  along the specific seam edge (see Fig. 11). We can formulate an energy that incorporates the symmetric fit along seams very similarly to (3):

$$\begin{aligned} E_{\text{rot+refl}} = \sum_{S \in \mathcal{S}} w_S \sum_{(e, e') \in S} l_e & (\angle(\mathbf{o}_e, \mathcal{R}_r(\mathbf{o}_e^{\text{rot}}, \mathbf{n}_e, k_e^{\text{rot}})) \\ & + \angle(\mathbf{o}_e, \mathcal{R}_r(\mathbf{o}_e^{\text{refl}}, \mathbf{n}_e, k_e^{\text{refl}})))^2, \end{aligned} \quad (7)$$

and computing a new integer value  $k_e^{\text{refl}}$  in the same manner as  $k_e^{\text{rot}}$ . We now minimize the energy  $E_{\text{rot+refl}}$  by iteratively taking the average of the closest integer rotations of  $\mathbf{o}_e^{\text{rot}}$  and  $\mathbf{o}_e^{\text{refl}}$  and adjusting Eq. (4):

$$\mathbf{o}_e \leftarrow \mathcal{R}_r(\mathbf{o}_e^{\text{rot}}, \mathbf{n}_e, k_e^{\text{rot}}) + \mathcal{R}_r(\mathbf{o}_e^{\text{refl}}, \mathbf{n}_e, k_e^{\text{refl}}). \quad (8)$$

We show in Fig. 12 (c) the resulting symmetric alignment of the wallpaper along seams.

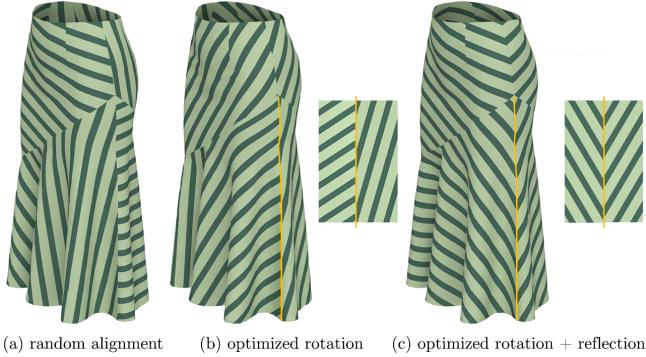


Fig. 12. The influence of reflectional symmetry along the seams on a skirt with 4 pattern pieces and a simple wallpaper of group PMM. A random orientation of the wallpaper is clearly visible and unpleasant to the eye (a). Optimizing for rotational fit minimizes the angle differences of the wallpaper along seams, but if the angle between the wallpaper and the seam is not exactly symmetric or equal on both sides of the seam, even a small angle difference is immediately apparent, and a perfect fit is impossible (b). Allowing for reflectional symmetries along seams results in visually pleasing alignments (c).

#### 4.2 Translational alignment

The rotation optimization in the previous step prepares the ground to improve the fit along seams, but since it does not contain positional information, it does not directly minimize our total fit energy in Eq. (1). We now fix the rotations and minimize the total fit energy by varying the translations  $t_1..t_{N_p}$ .

We employ the same strategy as for the rotational fit optimization and iteratively update the translation vectors  $t_c$  for all pattern pieces  $P_c$  to counteract the mismatch vectors  $d(v_e)$  along the seam edges:

$$t_c \leftarrow t_c + \frac{1}{w_c} \sum_{S \in S(P_c)} w_S \sum_{(e, e') \in S} l_e d(v_e), \quad (9)$$

with  $w_c = \sum_S w_S \sum_{(e, e')} l_e$ . We again iterate over all pattern pieces until our solution for the translation vectors  $t_c$  converges, in the same manner as described for the rotational alignment.

#### 4.3 Intermediate results

Finding the best alignment (both rotational and translational) of a wallpaper for each pattern piece is a global, highly non-convex optimization problem. Therefore, we cannot guarantee a global minimum but instead find a good local minimum starting from a given initial guess. Since we only optimize for a small number of variables (equal to three times the number of pattern pieces  $N_p$ ), our approach is fast and takes less than a second. This leaves us with several choices users could make. If they intend to find a best possible rotational fit, we run the above algorithm many times from different random starting points, choosing the result with the smallest energy and thus approximating a global minimum further. We found in our experiments that the quality of the local minimum depends on the wallpaper group, and especially for the groups PGG, CMM, P4, P4M, P3, P6 and P6M, repeating the process about 15 times starting from random wallpaper alignments further improves the fit. The

corresponding computation takes under 5 seconds for 15 repetitions, depending on the number of rotational symmetries, the amount of equivalent points in a basic cell and the number of seam edges. Secondly, we allow the user to alter the alignment by hand, creating a custom starting point and finding a close local minimum. If the user wishes to keep the chosen alignment of a specific pattern piece fixed, we simply omit this pattern piece in our iterations. It is also possible to optimize rotational and translational alignment independently, in order to set one of those by hand, and to interactively set seam weights. We show aligned results as intermediate steps in Figures 1, 3, 15 (left) and 16. Table 1 lists the energy values  $E_{fit}$  before and after alignment, and  $d(v_e)$  is visualized along seams in the 2D patterns in Fig. 1 and the 3D garments in Fig. 16. Though all examples show a great improvement in fit, they also show the main limitation of this approach: due to the shape of their seams (as explained in Fig. 9), most garments do not allow for a perfect fit (as in Fig. 15 (left)). Often the remaining misalignment along seams is numerically small but nevertheless obvious to the human eye, which easily notices discontinuities and mismatches in patterns.

### 5 SEWING PATTERN OPTIMIZATION

In this section we describe how we alter the sewing pattern shape to further improve the fit. In a first step, for each individual seam we compute a pair of new ideal seam lines in the 2D pattern space (see Fig. 14, top right), along which we would achieve a perfect fit of the wallpaper, but which are also possibly disconnected from other seam lines. In a second step, we deform the pattern shape towards the ideal seam lines by minimizing a bespoke energy (Fig. 14, bottom right).

*Ideal seam lines.* The computation of the ideal seam lines is illustrated in Fig. 13 and needs to adhere to two principles: for each vertex and each edge on the seam,

- (1) the twin 2D vertex coordinates  $p^{2D}$  and  $p^{2D,2}$  need to lie on equivalent points of the wallpaper;
- (2) the twin 2D edges  $e_k$  and  $e'_k$  need to have the same length.

The first principle ensures a perfect fit of the wallpaper and the second keeps pairs of seam lines consistent in length, such that they can still be sewn together. Denote the twin seam lines in 2D by  $S_1$  (on pattern piece  $P_i$ ) and  $S_2$  (on pattern piece  $P_j$ ). We first define a perfect shape for  $S_1$  while keeping  $S_2$  fixed. We do so by generating a number of candidate seams to replace  $S_1$ , which all fulfill both principles above, by applying the wallpaper symmetries to the fixed seam line  $S_2$  as described below. After choosing the best candidate  $S_1^*$ , we distribute the displacements between  $S_1$  and  $S_1^*$  equally between  $S_1$  and  $S_2$  to avoid drastic shape changes in the sewing pattern.

We create the best candidate seam  $S_1^*$  in four steps from the fixed seam line  $S_2$ , such that in each step the vertices of the altered seam line lie on equivalent points to the vertices of  $S_2$ .

- We move the seam curve  $S_2$  from  $P_j$  to  $P_i$  according to the wallpaper rotations  $R_i, R_j$  and translations  $t_i, t_j$  of the pattern pieces  $P_i$  and  $P_j$ , respectively.
- We apply the symmetries of the wallpaper, which create the  $n$  equivalent points of the basic cell, to the whole seam line, in

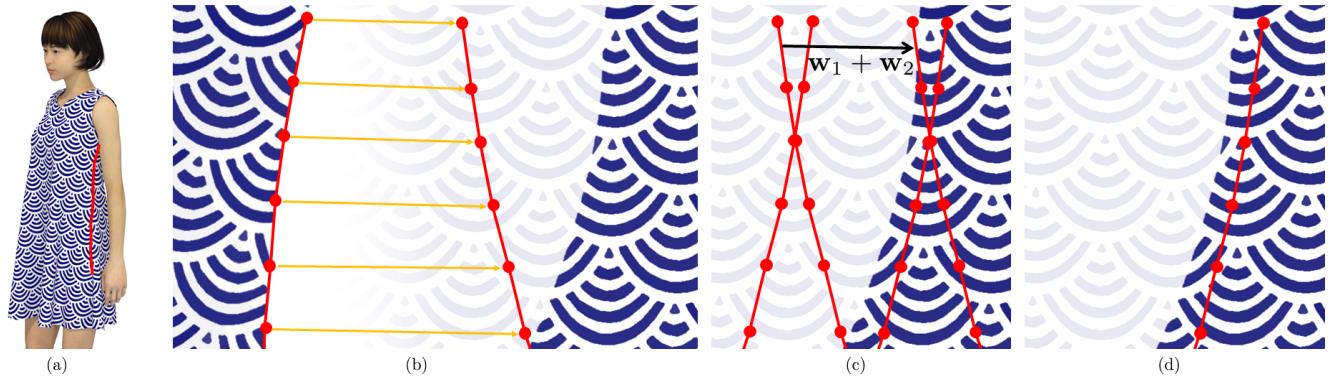


Fig. 13. Finding a perfectly matching seam line for a wallpaper of group CM. Starting from an imperfectly fitted seam (a), we first move one of the seam lines into the wallpaper space of its twin (b). For this wallpaper group, two types of perfect seam lines can be created by applying the wallpaper’s symmetries. All candidate seams are then moved as close as possible towards the second seam line by adding integer multiples of the vectors  $w_1$  and  $w_2$  (c). We choose the candidate that has the smallest distance to the original seam line (d).



Fig. 14. Pattern shape optimization. In the top row we show yoga pants and their sewing pattern with the wallpaper of group P4M already aligned using the method in Sec. 4. Overlaid on the sewing patterns are the disconnected ideal seam lines that would create a perfect fit. In the bottom row we show the optimized sewing pattern shape and the corresponding new yoga pants. The mismatch along the seams mostly disappears. Some small discontinuities of the wallpaper cannot be avoided, since the deformation approach only approximates the perfect seam lines.

order to create equivalent vertices of different type, resulting in  $n$  candidate seam lines. For example, using the wallpaper group CM would result in two candidate seam lines (see Fig. 5).

- We translate each candidate seam line as close as possible to the seam line  $S_1$  by applying integer multiples of the vectors  $w_1$  and  $w_2$ , keeping the type of equivalent points of each candidate seam line the same.
- We choose the candidate seam that lies closest to  $S_1$ , i.e., the one that minimizes  $\sum_{p^{2D} \in S_1} \|p^{2D} - p^{2D, 2, altered}\|^2$ .

The resulting curve  $S_1^*$  is an ideal seam line replacing  $S_1$ , perfectly fitting with seam line  $S_2$  and therefore fulfilling principle (1). Since all four steps are always applied to a whole seam line and none of the steps changes edge lengths, principle (2) is fulfilled as well.

To keep the deformation of the sewing pattern pieces small, we split the displacement between  $S_1$  and  $S_1^*$  in half and adjust both seam lines  $S_1$  and  $S_1^*$  equally, instead of a more drastic substitution of  $S_1$  by  $S_1^*$  alone. More specifically, we rigidly align  $S_1$  with  $S_1^*$  using Procrustes (see the review in [Sorkine-Hornung and Rabinovich 2016]) and then apply half of the rotation and translation to both seam lines  $S_1$  and  $S_2$ , yielding  $S_1^{\text{ideal}}$  and  $S_2^{\text{ideal}}$ .

*Shape optimization.* In order to approximate the computed ideal seam lines, we formulate the following energy:

$$\begin{aligned} E_{\text{pattern}} = & E_{\text{prox}}(\mathcal{S}) + \alpha E_{\text{conf}}(V_{2D}) + \\ & + \beta \sum_{f \in F_{2D}} \phi_f(A(f)) + \frac{1}{2\gamma} \sum_{S \in \mathcal{S}} w_S \sum_{(e, e') \in S} (c_e(S))^2. \end{aligned} \quad (10)$$

The scalars  $\alpha$ ,  $\beta$  and  $\gamma$  are weights. The first term  $E_{\text{prox}}$  is a simple proximity energy that encourages each seam line  $S$  to move towards the ideal shape  $S^{\text{ideal}}$ , as computed above:

$$E_{\text{prox}}(\mathcal{S}) = \sum_{S \in \mathcal{S}} w_S \sum_{p \in S} \|p - p^{\text{ideal}}\|^2. \quad (11)$$

The second term  $E_{\text{conf}}(V_{2D})$  is a conformal energy of the deformation that measures the triangles’ angular distortion using the cotangent Laplacian of the 2D mesh of the pattern pieces (see [Mullen et al. 2008] for further details). This term’s usefulness is twofold: Since we only adjust boundary vertices, it distributes the deformation

evenly over all triangles instead of only stretching near the boundary, and secondly, it improves the quality of the mesh, reducing the convergence time of the energy optimization.

The third term prevents triangle flips, which might occur due to overlapping ideal seams. It is a barrier term described in detail in [Schüller et al. 2013], where  $A(f)$  is the area of triangle  $f$  and  $\phi_f$  is the barrier function of that triangle.

The last term adds soft constraints to ensure that the lengths of twin edges remain equal. It is formulated through individual constraints for each edge  $e = (\mathbf{v}_m^{2D}, \mathbf{v}_{m+1}^{2D})$  and its twin  $e' = (\mathbf{v}_m^{2D,2}, \mathbf{v}_{m+1}^{2D,2})$  (see Fig. 6) as

$$c_e(S) = \|\mathbf{v}_m^{2D} - \mathbf{v}_{m+1}^{2D}\|^2 - \|\mathbf{v}_m^{2D,2} - \mathbf{v}_{m+1}^{2D,2}\|^2.$$

We use a quadratic penalty method to minimize this term, described in [Wright and Nocedal 1999]. The parameters  $\alpha$  and  $\beta$  are set to 1 for all our examples. We iterate starting with  $\gamma = 1$  and half it with each subsequent iteration, thereby increasing the weight until the constraints are numerically satisfied, meaning that the sum of all differences in length between twin edges is  $L := \sum_e |l_e - l_{e'}| < \epsilon$ . We found that setting  $\epsilon$  to half of the initial seam length difference already present in the 2D pattern of the input garment works well. The remaining, small length mismatches disappear in the 3D simulated garment due to the stretching of the fabric. In some rare cases  $L < \epsilon$  cannot be fulfilled (as in the case of the tunic, see Table 1) and we additionally restrict the number of iterations over  $\gamma$  to 20.

We use L-BFGS [Nocedal 1980] to solve the overall energy minimization Eq. (10). In the final step we use the resulting 2D pattern to re-simulate the 3D garment.

## 6 RESULTS AND DISCUSSION

We implement and test our method on a standard laptop and use 9 different garment models in our experiments, which we obtained from public sources. The sewing pattern of the red romper in Fig. 16 and the complex dress in Fig. 7 were published along with the work by Li et al. [2018]. The yoga pants in Fig. 14 and the paneled skirt in Fig. 16 are part of the Marvelous Designer Garment Library [2019]. All other garments were provided alongside the work by Brouet et al. [2012].

We refer to Sec. 4.3 for a description of intermediate results by alignment optimization alone. The final pattern shape optimization described in Sec. 5 takes 5 to 100 seconds with our unoptimized code, depending on the resolution of the input mesh (see Table 1). We illustrate our overall method on several garments in Fig. 16. The computed ideal seam lines are close to the original ones in all these examples, resulting in only slightly deformed sewing patterns, so that the final 3D garment does not visibly differ from the input garment (see also Fig. 14). As visualized by the pseudocoloring of the seams in Figures 1 and 16, the pattern shape optimization further improves the fit along seams. Note that for the red romper and the blue tunic in Fig. 16 we use the option to set the weight for the inconspicuous inner leg seams and the backwards facing shoulder seams to zero in order to improve the fit on other seams.

Table 1. For the garments named at the top, we list several attributes, including the energy  $E_{\text{fit}}$  for a random and optimized alignment and the optimized sewing pattern. In the middle section, we list attributes concerning the wallpaper alignment optimization, and in the bottom part, we list those concerning the sewing pattern optimization.

	Shirt	Dress	Romper	Skirt	Tunic
Figure	1, 3	16	16	16	16
faces	6k	9.5k	5k	119k	11k
$E_{\text{fit}}$ random [ $\text{cm}^2$ ]	298.2	359.9	215.5	314.4	366.9
$E_{\text{fit}}$ aligned [ $\text{cm}^2$ ]	122.2	225.6	149.7	149.0	230.3
$E_{\text{fit}}$ optimized [ $\text{cm}^2$ ]	22.4	100.8	123.4	38.2	120.4
<i>R</i> : avg. iterations	4	7	11	12	7
<i>R</i> : random samples	1	15	15	15	15
<i>t</i> : avg. iterations	20	20	23	39	8
<i>t</i> : random samples	1	1	10	10	1
initial $L$ [ $\text{cm}$ ]	0.3	7.5	13.6	2.0	3.1
final $L$ [ $\text{cm}$ ]	0.1	3.3	6.2	0.6	5.7
iterations over $\gamma$	18	13	14	20	20
total iterations	274	137	605	314	327
comp. time [sec]	10	8	25	99	20



Fig. 15. A failure case where the combination of a wallpaper with few symmetries and a garment with twin seam lines of very different shape result in visible differences in the 3D shape. On the left we show the input garment and pattern with optimized wallpaper alignment and on the right the optimized pattern and corresponding garment.

## 7 CONCLUSIONS

We have developed a method to align repetitive textile patterns along garment seams based on the theory of the 17 wallpaper groups, and furthermore an algorithm to deform the sewing pattern shape to further improve fit along seams.

*Limitations.* For certain garment and wallpaper combinations, the deformation of the optimized pattern might be significant enough to cause visible changes in the 3D garment shape, which can be problematic in tight-fitting garments. This usually occurs when the wallpaper group exhibits only a few symmetries (like the group P1) and at the same time the shape of the garment's twin seam lines does not fit well together (as illustrated in Fig. 15). In general,

based on our experimentation, we conclude that a higher number of equivalent points in the basic cell of a wallpaper and a smaller basic cell in relation to the overall garment size increase the quality of the fit achievable by aligning the wallpaper as described in Sec. 4, and therefore reduces the necessary deformation when optimizing the pattern with the method in Sec. 5. Our method is limited to repetitive patterns that do not exhibit any random variations, and we would like to extend our approach to other types of textile prints in the future, such as randomized dot patterns or animal prints. We currently do not take into account fabrication constraints of the sewing pattern (e.g., maximal allowed curvature of a seam).

**Future work.** In the future, we would like to incorporate the symmetries of the overall sewing pattern as well, such that pieces like sleeves of a shirt are guaranteed to be symmetrically deformed and textured. Further incorporating knowledge about the 3D simulation of the garment deformation into the pattern shape optimization step would allow us to directly control the changes in the 3D garment shape stemming from the pattern deformation. An automated weighting of the seams could be incorporated by using the view-independent visibility measure developed by [Zhang and Turk 2002]. In order to cut the pattern pieces with a laser cutter or a CNC machine, we would like to automatically detect the wallpaper group and orientation of a textile placed in the machine and compute the cutting paths for the pattern pieces. Optimizing the packing of the pattern pieces would allow us to reduce fabric waste.

## ACKNOWLEDGMENTS

We are grateful to Michael Rabinovich, Oliver Glauser, Christian Schüller, Shihao Wu and Alex Ion for invaluable discussions and help with the video, Rafael Wampfler for shirt examples and Magenta Zeng for textured cloth swatches. We thank Igarashi Lab for hosting the first author’s visit while working on this project.

## REFERENCES

- Noam Aigerman, Roi Poranne, and Yaron Lipman. 2015. Seamless surface mappings. *ACM Trans. Graph.* 34, 4 (2015), 72.
- Aric Bartle, Alla Sheffer, Vladimir G Kim, Danny M Kaufman, Nicholas Vining, and Floraine Berthouzoz. 2016. Physics-driven pattern adjustment for direct 3D garment editing. *ACM Trans. Graph.* 35, 4 (2016).
- Floraine Berthouzoz, Akash Garg, Danny M Kaufman, Eitan Grinspun, and Maneesh Agrawala. 2013. Parsing sewing patterns into 3D garments. *ACM Trans. Graph.* 32, 4 (2013).
- Remi Brouet, Alla Sheffer, Laurence Boissieux, and Marie-Paule Cani. 2012. Design preserving garment transfer. *ACM Trans. Graph.* 31, 4 (2012), Article-No. CLO. 2018. clo3d.com. <https://www.clo3d.com>.
- Philippe Decaudin, Dan Julius, Jamie Wither, Laurence Boissieux, Alla Sheffer, and Marie-Paule Cani. 2006. Virtual garments: A fully geometric approach for clothing design. *Comput. Graph. Forum* 25, 3 (2006), 625–634.
- CG Elves. 2019. Marvelous Designer Garments Library. <https://cgelves.com/shop/marvelous-designer-garments-library-incl-patterns-fabric-presets/>.
- ES Fedorov. 1891. Symmetry in the plane. *Zapiski Imperatorskogo S. Peterburgskogo Mineralogicheskogo Obshchestva [Proc. S. Peterb. Mineral. Soc.]* 2, 28 (1891), 345–390.
- Yuki Igarashi, Takeo Igarashi, and Hiromasa Suzuki. 2008. Automatically adding seam allowance to cloth pattern. In *ACM SIGGRAPH 2008 posters*. ACM, 15.
- Wenzel Jakob, Marco Tarini, Daniele Panozzo, and Olga Sorkine-Hornung. 2015. Instant Field-Aligned Meshes. *ACM Trans. Graph.* 34, 6 (Nov. 2015). <https://doi.org/10.1145/2816795.2818078>
- Michael Keckisen, Matthias Feurer, and Markus Wacker. 2004. Tailor tools for interactive design of clothing in virtual environments. In *Proceedings of the ACM symposium on Virtual reality software and technology*. ACM, 182–185.
- Tsz-Ho Kwok, Yan-Qiu Zhang, Charlie CL Wang, Yong-Jin Liu, and Kai Tang. 2016. Styling evolution for tight-fitting garments. *IEEE Trans. Vis. Comput. Graph.* 22, 5 (2016), 1580–1591.
- Minchen Li, Alla Sheffer, Eitan Grinspun, and Nicholas Vining. 2018. FoldSketch: Enriching Garments with Physically Reproducible Folds. *ACM Trans. Graph.* 37, 4 (2018). <https://doi.org/10.1145/3197517.3201310>
- Yanxi Liu, Robert T Collins, Yanghai Tsin, et al. 2004. A computational model for periodic pattern perception based on frieze and wallpaper groups. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 26, 3 (2004), 354–371.
- Yanxi Liu, Hagit Hel-Or, Craig S Kaplan, Luc Van Gool, et al. 2010. Computational symmetry in computer vision and computer graphics. *Foundations and Trends® in Computer Graphics and Vision* 5, 1–2 (2010), 1–195.
- Shufang Lu, PY Mok, and Xiaogang Jin. 2017. A new design concept: 3D to 2D textile pattern design for garments. *Computer-Aided Design* 89 (2017), 35–49.
- Niloy J Mitra, Mark Pauly, Michael Wand, and Duygu Ceylan. 2013. Symmetry in 3D geometry: Extraction and applications. *Comput. Graph. Forum* 32, 6 (2013), 1–23.
- Patrick Mullen, Yiyi Tong, Pierre Alliez, and Mathieu Desbrun. 2008. Spectral conformal parameterization. *Comput. Graph. Forum* 27, 5 (2008), 1487–1494.
- Rajkishore Nayak and Rajiv Padhye. 2017. *Automation in Garment Manufacturing*. Woodhead Publishing.
- Jorge Nocedal. 1980. Updating quasi-Newton matrices with limited storage. *Mathematics of computation* 35, 151 (1980), 773–782.
- C. Robson, R. Mahrik, A. Sheffer, and N. Carr. 2011. Context-Aware Garment Modeling from Sketches. *Computers & Graphics (Proc. SMI 2011)* 35, 3 (2011), 604–613.
- Kenneth Rose, Alla Sheffer, Jamie Wither, Marie-Paule Cani, and Boris Thibert. 2007. Developable surfaces from arbitrary sketched boundaries. In *Proc. Symposium on Geometry Processing*. Eurographics Association, 163–172.
- Christian Schüller, Ladislav Kavan, Daniele Panozzo, and Olga Sorkine-Hornung. 2013. Locally Injective Mappings. *Comput. Graph. Forum* 32, 5 (2013), 125–135.
- Olga Sorkine-Hornung and Michael Rabinovich. 2016. Least-squares rigid motion using SVD. Technical note.
- Emmanuel Turquin, Jamie Wither, Laurence Boissieux, Marie-Paule Cani, and John F Hughes. 2007. A sketch-based interface for clothing virtual characters. *IEEE Computer Graphics and Applications* 27, 1 (2007).
- Christopher W Tyler. 2003. *Human symmetry perception and its computational analysis*. Psychology Press.
- Nobuyuki Umetani, Danny M Kaufman, Takeo Igarashi, and Eitan Grinspun. 2011. Sensitive couture for interactive garment modeling and editing. *ACM Trans. Graph.* 30, 4 (2011), 90–1.
- Amir Vaxman, Marcel Campen, Olga Diamanti, Daniele Panozzo, David Bommes, Klaus Hildebrandt, and Mirela Ben-Chen. 2016. Directional field synthesis, design, and processing. *Comput. Graph. Forum* 35, 2 (2016), 545–572.
- Pascal Volino, Frederic Cordier, and Nadia Magnenat-Thalmann. 2005. From early virtual garment simulation to interactive fashion design. *Computer Aided Design* 37, 6 (2005), 593–608.
- Martin Franz Stephan von Gagern. 2008. *Computergestütztes Zeichnen in den Symmetriegruppen der euklidischen Ebene*. Master’s thesis. TU Munich.
- Tuanfeng Y. Wang, Duygu Ceylan, Jovan Popovic, and Niloy J. Mitra. 2018. Learning a Shared Shape Space for Multimodal Garment Design. *ACM Trans. Graph.* 37, 6 (2018), 1:1–1:14. <https://doi.org/10.1145/3272127.3275074>
- Amy Wibowo, Daisuke Sakamoto, Jun Mitani, and Takeo Igarashi. 2012. DressUp: a 3D interface for clothing design with a physical mannequin. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction*. ACM, 99–102.
- Stephen Wright and Jorge Nocedal. 1999. *Numerical optimization*. Springer Science.
- Eugene Zhang and Greg Turk. 2002. Visibility-guided Simplification. In *Proceedings of the Conference on Visualization ’02 (VIS ’02)*. IEEE Computer Society, Washington, DC, USA, 267–274. <http://dl.acm.org/citation.cfm?id=602099.602140>

## A TRANSFORMATION MATRIX $F_e$

We calculate the 3D orientation vectors  $\mathbf{o}_e = F_e \mathbf{r}_i$ , where  $F_e$  encodes the deformation of the triangle of pattern piece  $P_i$  incident on seam edge  $e$ . We can define a local basis of  $\mathbb{R}^3$  using the triangle edges  $\mathbf{e}_1^{3D}, \mathbf{e}_2^{3D}$  and the normal  $\mathbf{n}^{3D}$  of this triangle in the 3D garment space. Similarly,  $\mathbf{e}_1^{2D}, \mathbf{e}_2^{2D}$  and  $(0, 0, 1)^T$  form a local basis using the edges of the 2D triangle. We can then build the matrix

$$F_e = \begin{bmatrix} \vdots & \vdots & \vdots \\ \mathbf{e}_1^{3D} & \mathbf{e}_2^{3D} & \mathbf{n}^{3D} \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \vdots & \vdots & 0 \\ \mathbf{e}_1^{2D} & \mathbf{e}_2^{2D} & 0 \\ \vdots & \vdots & 1 \end{bmatrix}^{-1} .$$

Note that we regard  $\mathbf{e}_1^{2D}, \mathbf{e}_2^{2D}$  and  $\mathbf{r}_i$  as 3D vectors with the third coordinate being zero.

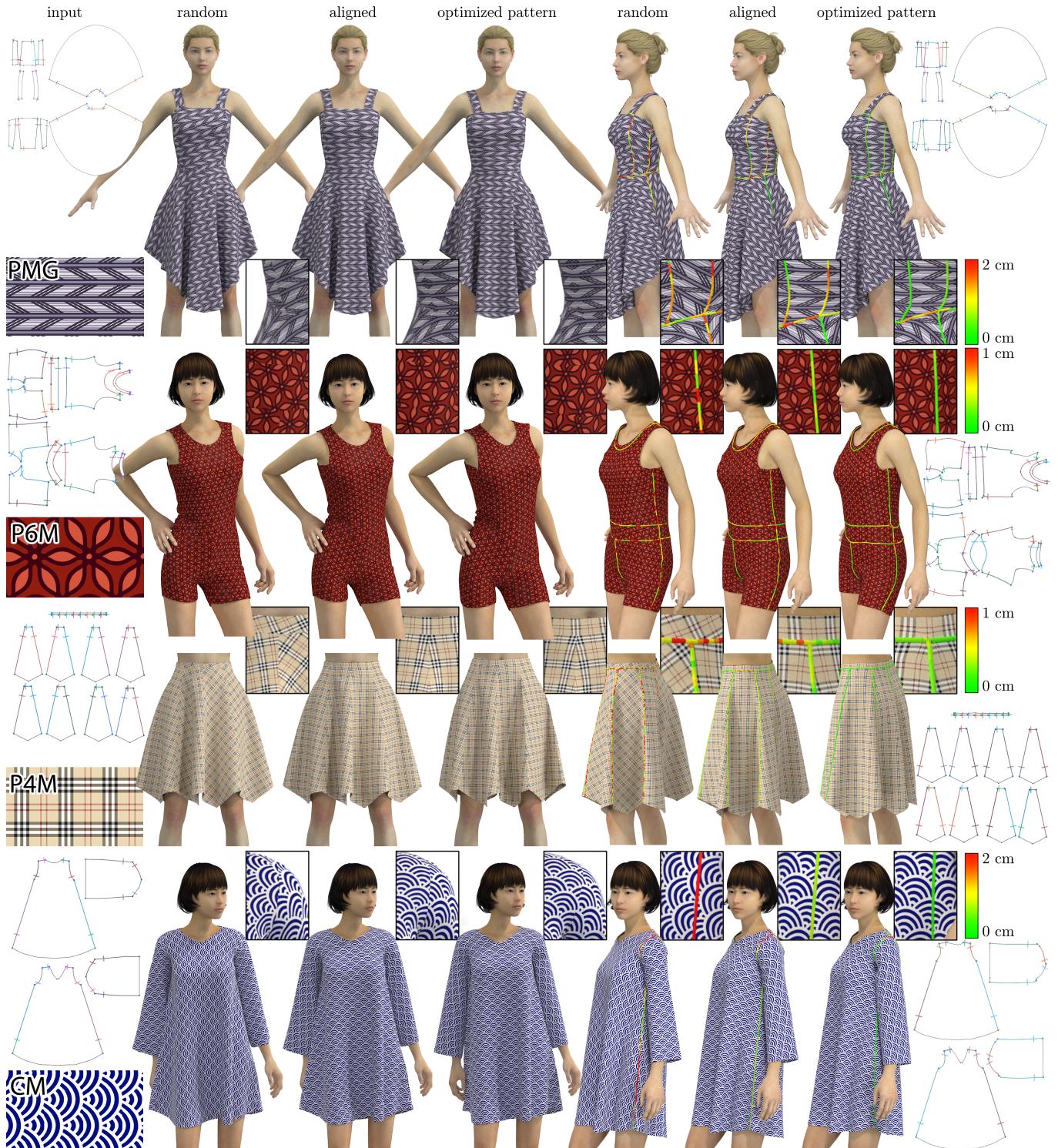


Fig. 16. Results for different garments and textures. On the left, we show the input sewing pattern and texture with the name of the wallpaper group. Next we show the front view of three different versions of the garment: The first shows a random alignment, the second shows an optimized alignment of the wallpaper for the input sewing pattern, and the third shows an optimal alignment for an optimized sewing pattern. The side view shows the same three versions of the garment. While the front view shows the unaltered overall appearance of the textured garment, we additionally colorize the mismatch  $d(v_e)$  for each seam edge in the side view, with green corresponding to a perfect fit and red to a mismatch of 2 cm for the top and bottom row, and 1 cm for both middle rows.