From Fiber to Fabric: Interactive Clothing for Virtual Humans

George Baciú* and Wingo Sai-Keung Wong†

Abstract

The virtual world is currently limited in the graphic representation and visualization of material designs. The limitation is in part due to the limited range and the relative simplicity of spectral properties of the materials simulated in a virtual environment. Our work starts with a fast classification and retrieval for handling the large numbers of multi-scale texture samples of complex deformable materials, such as, woven and knitted fabrics. We have developed a general system that can also serve as a unified platform for other material analysis and classification and a 3D panel design system for 3D clothing modeling and draping. Multi-scale color theme indexing for image acquisition and retrieval can be more intuitively supported by a multi-touch gesture interface that is now the preferred mode of interacting with tablets, screens, and other visual communication devices. These also add to the collaborative modes of input and retrieval in fabric and fashion design for virtual agents. In this paper we describe the process of texture analysis of real fabric materials and preliminary models for the 3D texture generation for virtual clothing, color theme design, and ultimately 3D draping based on robust collision detection methods.

CR Categories: I.3.3 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

Keywords: virtual yarn, fabric, cloth, collision detection, draping, virtual agents

1 Introduction

Much has been done by researchers on virtual agents and virtual humans. The most comprehensive overview can be found in the ever-lasting collection of the best papers published over many years by Nadia Magnenat and Daniel Thalmann [Magnenat-Thalmann and Thalmann 2007]. However, the virtual world is still limited in the graphic representation and visualization of material designs. The limitation is in part due to the limited range and the relative simplicity of spectral properties of the materials simulated in a virtual environment.

Our work starts with a fast classification and retrieval for handling the large numbers of multi-scale texture samples of complex deformable materials, such as, woven and knitted fabrics. We have developed a general system that can also serve as a unified platform for other material analysis and classification. Multi-scale color theme indexing for image acquisition and retrieval can be more intuitively supported by a multi-touch gesture interface that is now the preferred mode of interacting with tablets, screens, and other visual communication devices. They also add to the collaborative

modes of input and retrieval in fabric and fashion design for virtual agents. In this paper we describe the process of texture analysis of real fabric materials and preliminary models for the 3D texture generation for virtual clothing, color theme design, and ultimately 3D draping based on robust collision detection methods.

Hundreds of thousands of fabric designs are produced and sourced annually in the textile industry. Storage, classification, and retrieval of these fabric designs have become a daunting challenge. A large percentage of fabric designs are misclassified or simply lost due to human error or lack of a systematic approach to indexing and retrieval of fabric designs. Fabric designs evolve over time. Verification and evaluation of various classes as well as individual patterns is an evolutionary process. The evolution of fabric design depends strongly on current material resources as well as innovation in material science. This is coupled with creativity in design, shape, color and ensembles. The texture appearance reflects the characteristics or state of the creativity and innovation process of design.

2 Texture analysis of fabric materials

Material texture is often generated as a combination of micro-structure feature and narrow band spectral reflections. In textile materials, fabrics and garments exhibit a wide range of spectral reflection variations under various lighting conditions that affect their appearance and use. Fabric designers are sensitive to a broad classifications of fabric designs. An important tool for fabric design is the multi-scale visualization of fabric texture that allows designers to classify and evaluate design patterns based on observations at different scales of resolution. Another tool is the multispectral response and appearance of fabric material under different lighting conditions. In the non-visible spectral range, infrared can be used to identify patterns of energy exchanges at the fiber level due to friction, strain, or electro-chemical factors. Most importantly, visualizing large collections of patterns in the context of cultural and demographic use of materials provides further depth of knowledge in the use of materials in various ethnical regions around the world. In the textile fabric design and manufacturing, representation and retrieval of patterns and texture designs is paramount for communicating both with clients as well as with production managers and designers.

Visualizing the combinations of structure and spectral variations of a fabric design before the implementation of the weaving plan is an important requirement and a necessity in the production cycles of textile fabrics. In this context, structural inference of high-level patterns (HLPs) is needed in order to build new interfaces for analyzing large scale material texture data, that is, in the presence of a large collection of imaging samples that may vary across different spectral bands and structural scale. The structure inference parses the detected repetitive patterns and forms higher-level abstract representations for texture images using generators. A generator is the basic unit of an arbitrary pattern organization. In our generator-based inference model, the generators are built from the extracted structure completion fields. The HLP completion fields indicate the connection orientations of each repetitive patch. We can assign these orientations to the generators and connect them in a pair-wise manner to form topological hyper-graphs of connected patterns. Based on the topological generators of these hyper-graphs we aim to examine the inference rules that generate such HLPs.

Many material surfaces as well as object ensembles arranged for

*e-mail:csgeorge@comp.polyu.edu.hk, GAMA Lab, Department of Computing, The Hong Kong Polytechnic University.

†e-mail:wingo.wong@gmail.com, Department of Computer Science, National Chiao Tung University, Taiwan, R.O.C.

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 commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
SA'14 Workshop on Autonomous Virtual Humans and Social Robot for Tele-presence, December 03, 2014, Shenzhen, China
Copyright © ACM 978-1-4503-3243-9/14/12 \$15.00
<http://dx.doi.org/10.1145/2668956.2668959>

Figure 1: Micrograph of real woven fabrics:(a) micron-level woven yarn; (b) multi-color yarn blending; (c) energy levels of 3D yarn.

decorative or functional purposes exhibit near repetitive structural compositions, for examples packages of items arranged in a store, tapestry, fabrics. Less obvious are the dynamic near repetitive patterns, such as flocks of birds, crowd patterns in congestion zones, traffic, etc. Most commonly found repetitive patterns are in woven fabrics.

The color attributes at the fiber level coupled with yarn density and weave patterns of fabrics provide a rich composition of texture that impact both the design evaluation as well as the production requirements in textile manufacturing. Other applications of material repetitive pattern analysis are pervasive through many other applications as for example medical imaging, crowd behavior, interior decoration, architecture and urban planning, agricultural zoning, etc. These are all higher-level patterns (HLPs) that can be analyzed, indexed and archived for future retrieval and use. In our current iTextile project we work with three leading garment manufacturers in material pattern analysis with the main objective to classify and archive in excess of 1,350,000 samples of existing physical fabric patterns and designs.

In the past six years, we have focused on multi-scale woven fabric analysis [1-29,84]. 1. We have developed methods for accurate fabric indexing [9], and evaluation [10]. The multi-scale representation is used in our new iTextile Database [2]. This provides an intelligent search and classification system to identify, prioritize woven fabric designs and their physical attributes. Currently, the database accommodates over 19,000 samples of real commercial fabrics with complete manufacturing attributes and design specifications. This allows designers to evaluate, modify, analyze, search, render and prototype fabric designs.

We have been working on image capture, analysis and 3D draping simulation of garments. We have developed a number of garment design tools as well as collision detection modules that robustly and efficiently handle high resolution 3D fabric simulations and multi-layer clothing panels [2-29]. Our high performance draping simulation platform can be used for interactive evaluation of textures and shapes on garment designs [11-13]. Fabric designs [5,9], modeling garment panels, and 3D deformable garments are supported on this platform. Furthermore, we have developed a garment panel shape matching engine for standard textile panels with consistent panel shape boundary matching for prototyping virtual garments [6].

The ultimate challenge in computational fashion and fabric design is the understanding of the cognitive processes [85] employed during the creative mode of designing textures and repetitive patterns. This work is motivated by the need to better understand the cognitive measures of higher level patterns that can be applied to the study of fabric design and the transmission of qualitative informa-

tion [86, 87] related to the perception of novelty and beauty. Understanding the connection between low level patterns and higher level compositions requires the extrapolation of texton theory to images of arbitrary ensembles of objects.

3 Generative textures

In Julesz’s texton theory [Julesz 1981; Julesz and Bergen 1983; Julesz 1984; Malik et al. 1999; Cai and Baciú 2011] the elements of texture perception are all unified into pattern units called textons. Contrary to the previous Gestaltists’ belief that texture is observed as a meaningful whole, in texton theory, the higher perception of structure is considered to be built upon low-level elements. Long after the initiation of texton theory in the early 1980s, the idea of texture units became pervasive in modern texture discrimination and texture segmentation. For example, the early work on texture boundary [Voorhees and Poggio 1988] texture discrimination [Malik and Perona 1990; Jain and Farrokhnia 1990] image segmentation [Malik et al. 1999] and object recognition [Jurie and Triggs 2005] were all derived from texton theory. However, the use of texton models in texture analysis or pattern recognition, as pointed out by Malik and Perona [Malik and Perona 1990] relies heavily on texton density rather than texton composition. In the density models, textons are often considered as special types of filters rather than invariant compositional units.

4 Near-regular textures

The first published work using the term ”repeated structures” is Mundy and Zisserman’s paper in 1994 [Mundy and Zisserman 1994]. The idea proposed was about the extraction of the invariance in 3D structure recovery. This was based on the structural correlation obtained from different views. Malik et al. [Malik et al. 1999] pointed out the importance of detecting repeated elements in structural analysis for an image. They provided a detection algorithm using a window matching and region growing framework. As an extension of the traditional approach, Hays et al. [Hays et al. 2006] applied higher order structural analysis to analyze the organization of texture. They also used a window matching algorithm, the normalized cross correlation (NCC), but designed a deformable and expandable lattice template to detect grid structures in images. Based on this initial work many other techniques have been developed including, correlation-based methods Hays et al. [Hays et al. 2006], hashing-based methods, Tuytelaars et al. [Tuytelaars et al. 2003] similarity composition for irregular pattern extraction, Irani and Boiman [Irani and Boiman 2007] templates of binary sketches, Bagon et al. [Bagon et al. 2010] geometric tem-

Figure 2: Analysis of near repetitive patterns

plate grouping and local symmetry based approaches using detected pairs of repetitive SIFT features [Huang et al. 2009] Local symmetry grouping was found to be effective in large repetitive structures in urban scenes, Wu et al. [Wu et al. 2010; Wu et al. 2011] Global symmetry grouping approach has been developed by Terzopoulos et al. [Terzopoulos et al. 1988], Zhao et al. [Zhao and Qian 2011], and Yang et al. [Yang et al. 2005].

5 Pattern alignment

Our local patch alignment attempts to resolve the repetitive detection problem based on a Lie-algebra induced constraint for aligning repeating structures in near regular textures. The refined geometric shapes of the repetitive patches are more adaptive in representing the distorted image structures. Our method focuses on local pattern alignment rather than global grouping. The local alignment may lose some efficiency in the detection, but gains flexibility in the analysis of pattern organizations.

6 Long-term challenges

The long-term challenges of this project is to infer local patterns from global repetitive pattern for use in design evaluation, symmetry detection, matching and detection of similar materials and material quality and to tie this information into a matching and retrieval process for manufactures of various products ranging from textiles, construction material, pulp and paper industry, food, environmental monitoring and urban planning. Repetitive pattern detection is widely used in the applications of structure reconstruction, for example texture structures reconstruction. The repetition detection is also highly observed in urban building recognition and building model recovery as well as in urban scene reconstruction.

7 Pattern group segmentation

Pattern group segmentation requires a new higher-order pattern representation model. In this project, we will use the generative texton model to represent repetitive patterns together with a Lie-algebra parameterization to describe the shapes and deformations of the pattern patches. The patterns will be explicitly represented as deformable local image patches, and the invariant texton extracted by the alignment of the local patches.

8 Generators for deformable structures

We use a locally deformable generative texton model as this has been shown to be highly effective in prior testing. The model is

based on a number of clustering methods, such as multidimensional scaling and the mean shift clustering. Our model described two conditional scenarios in the repetitive pattern detection. The first condition is the pattern reproduction. This condition ensures the repetitive patterns are generated by one specific texton. The repetitions are texton replicas with certain geometric transforms. The second condition is the pattern alignment. This condition requires that if two repetitive patterns are equivalent to each other, one pattern can be deformed geometrically to another. In a set of repetitive patterns, the alignment condition indicated that an invariant common pattern can be extracted from the joint-alignment for the patches. In our clustering model, the repetitive patterns are represented as image patches.

We developed a multi-dimensional scaling (MDS) embedding method [Cai and Baciú 2011] that provides an effective dimension reduction and produces an initial clustering for the data. Then the mean shift algorithm is used for further clustering the low-dimensional embedded data. The mean shift process allows us to smoothly merge similar patterns in the low-dimensional subspace, making the clustering more effective and robust. We propose a modified formulation for the mean shift process and develop new repetitive pattern detection for large scale images.

The grouping of detected patches is accomplished in the refinement step of the main algorithm. The task of refinement is to adjust the shapes of local patches among local area. For example, Figure 2 shows an iteration for grouping lower-level patterns into higher level ones.

By applying the higher-level segmentation, the locations and shapes of repetitive patterns from a texture image can be immediately obtained. However, the global texture organization needs to be recovered. In this project, we extend the local patch grouping to global structure inference. The inference attempts to discover (1) how the patches detected can be organized into a global structure, and (2) the type of the global structure.

9 Multiscale virtual fabric materials

In order to facilitate more accurate geometric representations for highly complex structures found in woven fabrics, we developed a new generator for fiber and yarn representations. In this new framework we can perform multi-scale rendering of complex fabric surfaces at interactive rates [Zhang et al. 2013].

Woven fabrics constitute the primary materials of casual and formal garments from common shirts, suits and dresses to silk ties, attachments and accessories as well as decorative covers for interior

Figure 3: Generated virtual woven fabric at three scales. (a) fiber scale; (b) yarn scale; (c) panel scale.

design. As such, woven fabrics are a source of rich and complex texture that is often defined by simple elements found at the level of fiber or yarn interlacing structure.

Furthermore, the technology and performance of weaving machinery have drastically improved the quality and density of woven fabrics. Hence, the requirements for greater verisimilitude in the woven details of fabrics for all kinds of applications, e.g. virtual design, structural simulation, and computer games. The design of woven fabrics for garment panels, generating realistic woven patterns in a 3D environment, is a functionality that is currently missing in all single scale fabric texture design systems used in the textile industry. This creates the need to visualize the scale-varying appearance of fabrics for virtual clothing and multi-scale rendering at interactive rates.

Figure 3 shows three scales at which woven fabrics are often viewed. At the micro scale, figure 3c, we can see the fiber construction of the yarn made up of intertwined strands. At the meso scale, figure 3b, the weave of the yarn is visible. At the macro scale, figure 3a, the cloth appears smooth.

Displaced subdivision surfaces (DSS) [Lee et al. 2000] provide an effective way to represent detailed surfaces. A DSS patch defines both the surface topology and parameterization by using a coarse control mesh and also captures the fine geometric detail using a scalar-valued function. The DSS representation also permits geometry separation by providing smooth domain surface and detail displacement fields. However, this is not sufficient to represent a detailed surface with scale-varying appearance such as woven fabrics.

10 Color theme-based design

Computer aided design tools for textiles, clothing and fashion have reached a high level of sophistication in design functionality especially with the advent of high precision weaving and knitting machines. However, some open problems are yet to receive attention in fashion design. These all share a common background, the understanding of the cognitive processes of creativity in color design. For example, in Figure 4 we observe a high correlation of ambient color tones that add to significantly to the aesthetic perception of beauty and elegance in design. More importantly where does the inspiration for mood and color assembly come from? Figure 5 shows a possible generation of color assemblies [Zheng et al. 2010; Zheng et al. 2012].

The emotive selection of color assembly is highly related to a spectral composition of gradients that can be found in natural scenes. This is indicative of color theme-based design which often has deep

cultural undertones and can express a season, contrasting shades even motion and energy.

The two most important aspects in achieving desired fabric designs are: (1) color spectrum selection, and (2) mood board presentation. These aspects are difficult to model computationally, but are fundamentally crucial tasks for both amateur and professional fashion designers.

We have worked on a new framework for fabric design that simultaneously takes into account the design of the fabric color and the texture effects in a cognitive loop by providing operations at the level of color theme assembly. In our framework, we cast the process of cognition in fabric design as an optimization problem for generating a suitable cognitive color theme design combined with fabric texture properties.

Our framework consists of a data-driven color theme optimization approach to facilitate color theme extraction, color mood clustering, and weave pattern entropy functions to isolate color themes from abstract or natural images associated with a subject or a theme. The measure of the difference between input images and fabric designs is quantified in a color mood space by using a color emotion metric. The output of the color theme is a modified version of the subject of the input images that is achieved by translating the weights from the color mood space into a semantic descriptor of the subject.

Fashion designers can also use the system to optimize their fabric designs according to the on-line color theme database, such as found on Kuler website and COLOURLovers. The system can automatically generate new fabric designs and present the designs in mood boards that rival complex commercial designs and presentation boards done by professional designers.

11 Realistic draping of virtual clothing with continuous collision detection

Deformable objects often take the shape of an underlying support structure. During this process, the objects may interact with each other and generate collision events at each simulation time step. For an accurate visual representation of the motion of these objects, their contact regions must be detected and resolved appropriately so that a penetration free state could be achieved without visual artifacts. A penetration free state can be achieved for deformable objects when continuous collision detection (CCD) is employed. A comprehensive survey on collision detection of deformable objects can be found in [Teschner et al. 2005]. In this section, we review the main state-of-the-art techniques for continuous collision detection of deformable objects and suggest potentially new directions in this area of study.

Figure 4: Color theme-based fashion design

There are two types of collision events: (1) self-collisions and (2) inter-collisions. Self-collision events occur for the same deformable object while inter-collision events occur between two different objects.

Bounding volume hierarchies. A common data structure, known collectively as hierarchies of bounding volumes, are often employed due to their effectiveness in culling objects that are considered to be far away from each other and have a low probability of collision during the next sampling time frame.

The types of bounding volumes are, for examples, spheres, axis-aligned bounding boxes [van den Bergen 1999], oriented bounding boxes, discrete polytopes [Klosowski et al. 1998], capsules and cylinders.

The hierarchies of bounding volumes are precomputed at the processing stage. Each node of a hierarchy is associated with a subhierarchy and it is assigned a bounding volume. A leaf node bounds a triangle. A common approach to build the BVH is to use the median splitting method.

At the runtime stage, each hierarchy of the bounding volumes is refitted so that the bounding volume bounds the swept volume of the primitives in the subhierarchy. As a deformable object deforms, the quality of its hierarchy of bounding volumes may degrade over time. Consider that N_r and N_b are the number of potentially colliding primitives collected by using the refitted BVH and the rebuilt BVH, respectively. If N_r is 'significantly' larger than N_b , then the quality of the refitted BVH degrades. In this case, the hierarchy of bounding volumes may be rebuilt in a selective manner. Please refer to [Teschner et al. 2005] for details.

Normal cone based methods A pure BVH-based method must perform the BVH traversal even though the object is free of self-collision. For example, the surface of a sphere that is free of collision. Volino proposed a normal cone based method to eliminate the check of a surface in discrete collision detection if it satisfies two conditions: (1) there exists a vector such that the dot product is positive, and (2) the contour of the surface is self-collision free. The method is extended to continuous collision detection in [Provot 1997; Mezger et al. 2003]. Different techniques have also been proposed for computing the continuous normal cones in [Wong and Baciú 2005a; Tang et al. 2009a]. These methods can be applied to continuous self-collision detection in general cases of deformable objects. The contours of such methods are precomputed for each internal node of the BVH.

Radial view based methods Wong et al. [2013] proposed a radial view based culling (RVBC) method for performing continuous self-collision detection of closed deformable meshes. The basic idea of the RVBC method is that if the facing direction of each triangle of the mesh is consistent with the viewing direction from a point to that triangle, then the mesh is free of self-collision. A dynamic clustering scheme was presented in [Wong et al. 2014] for improving the performance of the RVBC method. There are clusters that may form a convex shape and these clusters could be merged as a single cluster. The cost spent in inter-cluster checks would be reduced. Later on, Wong and Cheng [2014] presented a method to adapt the RVBC method to handle meshes with round shaped holes. Ghost triangles are added to fill the holes and then an extended mesh is obtained. The extended mesh is closed and 2-manifold so that the RVBC method can be applied on the extended mesh.

Low Level Culling. For a triangle pair, there are fifteen feature pairs to check, including six vertex-triangle pairs and nine edge pairs. However, some of these feature pairs are duplicated for triangle meshes because there are vertices and edges which may be shared by more than one triangle.

Wong and Baciú [Wong and Baciú 2006] suggested that the shared features could be assigned to attached triangles. This kind scheme performs a collision check for a feature pair once. However, it is only applicable for inter-collision detection. The bounding volume overlapping tests of feature pairs are effective in pruning away the non-colliding feature pairs as suggested by Hutter and Fuhrmann [2007]. Curtis et al. [2008] improved on the feature assignment scheme proposed in [Wong and Baciú 2006]. A compact data structure is adopted to extract the potentially colliding feature pairs of a triangle pair. Wong and Baciú [2014] proposed to group triangles into clusters and assigned features to clusters. They found that the performance of collision detection is good for grouping at most two triangles as a cluster.

Zhang and Kim [2012] proposed a set of pruning schemes to evaluate whether or not a feature pair collides. In some cases, although the global swept volumes of triangle pairs overlap, their relative swept volumes do not overlap. Wong et al. [2014] proposed to check relative AABBs of the potentially colliding triangles pairs before the elementary test processing.

Elementary test processing The purpose of the elementary test processing is to compute the contact times and other contact information of the potentially colliding feature pairs. Usually, the interval Newton method is employed to compute the roots of cubic poly-

Figure 5: *Color assemblies*

nomial equations as solutions of the collision point. It is reported that a large portion of the potentially colliding feature pairs (i.e. vertex-triangle and edge-edge) do not collide [Tang et al. 2010]. Tang et al. employed deforming non-penetration filters to prune away the feature pairs that do not become coplanar within the time interval. Tang et al. [2011a], applied parallel filters for the pruning purpose. Wong and Baciú [Wong et al. 2010] used the coefficients of a cubic equation to determine whether or not a feature pair collides. They employed an approximated method to compute roots of cubic equation. But this may lead to intersection problems of feature pairs.

The aforementioned methods do not consider the rounding errors and numerical errors. Thus, the feature pairs may intersect with each other at the computed contact times. Bridson et al. [Bridson et al. 2002] suggested the use of error tolerance when computing the contact times. However, it is not easy to assign a suitable value for the error tolerance for different simulation cases. Brochu and Bridson [2012] decomposed continuous collision detection into a set of simplex intersection tests to determine whether or not a feature pair collides. This method is effective in handling meshes that require re-meshing during simulation. Wang [2014] proposed a robust method to tackle the numerical problems and rounding errors when the contact times of feature pairs are computed. The development of the proposed method is based on error analysis.

11.1 Parallel collision detection techniques

There are techniques which are implemented on multicore CPUs and GPUs. A task can be divided into smaller independent tasks that can be executed in parallel, leading to tremendous speedups.

Wong and Baciú [2005b] employed GPUs to compute the roots of cubic equations and distances of feature pairs. The positions of vertices, topology data of meshes and potentially colliding triangle pairs are stored in textures. Kim et al. [2009] proposed a lock-free parallel algorithm. The BVH traversal is performed on CPUs while elementary tests are handled on GPU.

Tang et al. [2009b] adapted a front tracking approach in the proposed parallel algorithm. As such, the computation cost spent in BVH traversal is reduced. Pabst et al. [2010] developed a parallel technique for computing discrete and continuous external and self-collisions of deforming and rigid bodies. They suggested the use of simple coefficient checks on GPUs to eliminate non-colliding

feature pairs [Wong 2005]. Lauterbach et al. [2010] developed parallel algorithms which find collisions and compute separation distance between models. They used hierarchical structures of bounding volumes with shallower and higher branching factors compared with the traditional methods.

Tang et al. [2011b] adopted a deferred front tracking technique to improve the process of collision detection. A stream registration technique is employed to collect the potentially colliding triangle pairs. This technique stores the potentially colliding triangle pairs in one array after all the threads finish their own BVH traversal task. Tang et al. [2013] proposed to map the entire cloth simulation process on GPUs, including collision detection, time integration and collision response processes.

12 Draping and detail enhancement

The purpose of cloth draping simulation is to quickly bring cloth to a rest state. This is particularly important for generating a stable cloth after the panels of the cloth are sewn together. The major problem is that at the time the panels are connected together, there are adjacent vertices which are too far away from each other. If a direct integration approach is adopted, such as explicit Euler integration, the system would be numerically unstable. Of course, a small simulation time step or implicit integration methods could be used to make the system stable. However, the cloth draping system would take a long time to converge in this case.

Wong [2005] applied a piecewise stable sewing process to dynamically adjust the movement speeds of the vertices in cloth draping simulation. The cloth models are spring-mass systems. The velocities of vertices are reset at some moments to reduce the kinetic energy of cloth. The velocities of the particles are set to zero in the end of the sewing process or when the potential energy is increasing. Several other schemes are proposed and they are compared with each other. Volino and Magnenat-Thalmann [2007] adopted a similar approach called *stop-and-go* for cloth draping simulation. If the total potential energy reaches a minimum, the velocities of all vertices are set to zero. Kang and Lee [2007] proposed to decouple the cloth draping simulation into two parts: global shape computation and local detail enhancement (i.e., local wrinkles). The cloth is assumed to be sewn already. These methods require physics based simulation at runtime and thus the overall draping process is slow. Umetani et al. [2011] proposed a draping system that enables the

editing of the panels with low complexity at real time rates. Furthermore, users can also directly manipulate the 3D garments, e.g., changing the size of a part of a garment. The simulated results are produced interactively.

Kavan et al. [2011] proposed an upsampling method to add details (e.g., wrinkles) to the coarse mesh of cloth under a specific context (e.g., wind blowing). Guan et al. [2012] proposed a system called DRAPE which stores a model about the simulation results of clothing on bodies of different shapes and poses. Given an input of a human body with known shape and pose parameters, the proposed system can fit a garment with wrinkles on the human body. Although it takes some time for the system to learn from the physics based simulation, its runtime performance is at real time rates. Hahn et al. [2014] computed a set of basis vectors for the simulation data of cloth at the preprocessing stage. Given a human body with the pose information, a subset of the basis vectors is selected to generate detailed folding patterns of cloth. Although these techniques are fast, they need the complete garment information. They do not handle the sewing process.

13 From pattern sewing to draping process

Given a body mesh M , pose P and a set of patterns T . The goal is to sew the patterns and produce the final garment dressed on the human body (see Figure 6). The body mesh M is a triple (F, V, E) , where F is the set of faces, V is the set of vertices and E is the set of edges E . A pose P is a tuple (S, J) , where S is the skeleton of the body mesh and J is the set of orientations of joints of skeleton S . The set of patterns, $T = \{t_1, t_2, \dots, t_n\}$, contains n patterns. Each pattern contains a mesh and styling elements such as darts, hemlines and pleats.

Figure 6: An illustration from sewing patterns to draping process. (a),(b): there are six patterns which are aligned with the body mesh M . (c): the initial garment mesh $M_G^{initial}$ after applying the stitching information for patterns. (d): the garment mesh G at rest state after draping simulation.

There are several challenging issues that should be handled: 1) the stitching information C for T ; 2) the alignment between M and T ; 3) initial 3D garment mesh $M_G^{initial}$ obtained from T with stitching information; and 4) draping simulation of $M_G^{initial}$ to obtain mesh M_G at the rest state. We use a simple example as shown in Figure 7 and Figure 8 to illustrate these four challenging issues.

Stitching information. Initially, we only have the patterns but we

Figure 7: An illustration from sewing patterns to draping process. (a): patterns and a body model. We show some thick blue lines which indicate the sewing lines (right hand side). (b): a triangulation of the left sleeve pattern. (c): The meshes of the patterns and the body mesh in 3D space.

do not know their edge stitching correspondences C . Of course, if a garment designer designs the patterns, he or she should know C . However, to make up the information about C , it would be time consuming and tedious to do so. For example, an animator may only design the patterns or even extract them from a database. Then he or she wants to directly perform the draping simulation for the patterns to create a garment mesh M_G . The situation gets worse if the patterns are changed often. In this case, a lot of time would be spent in setting up the edge stitching correspondences C . Thus, it is desirable to semi-automatically generate C . Indeed, it would be able to reduce the design cycle of garment design.

Berthouzoz [2013] developed an automatic system for parsing sewing patterns and converting them into 3D garment models. An algorithm based on machine learning and integer programming is proposed to compute the stitching edge correspondences between patterns. The authors reported that their approach achieves 68% correctness. It would be desirable to pre-compute a database which contains a variety of stitching patterns. Then, it would be able to apply machine learning techniques to generate C with higher accuracy. Figure 7(a) shows that the contour of the left sleeve pattern is split into two parts, one mapped to the front pattern and another mapped to the back pattern. It is difficult to solely rely on lengths of contours for computing the edge stitching correspondences.

Alignment between patterns and body mesh M . We assume that the body mesh M is an open armed model, as shown in Figure 7(a). We consider that each pattern is associated with a name tag. Based on the name tag, an initial seed position can be computed [Berthouzoz et al. 2013]. The idea is that a name tag such as 'left sleeve' refers to that the corresponding pattern is placed near to the left arm. Thus, a lookup table can be built to associate the name tags with the parts of the body model. This requires a standard way

Figure 8: (a): The white lines are generated based on the stitching edge correspondences of patterns. If the orientation of the mesh of the left sleeve is incorrect, the white lines intersect with the meshes. (b): The orientation of the mesh of the left sleeve is correct. (c): A mesh is generated based on the stitching edge correspondences of patterns.

to define the name tags. The question is that *if the patterns have incorrect name tags or do not have name tags, how the alignment process can be performed efficiently?* This happens for people who do not have a standard training. We consider that a garment would tightly cover the body and it is symmetric. Based on the symmetry property of the garment and the body, we may be able to compute an initial seed position for each pattern. This can be done as follows. As we know the edge stitching correspondences C , the patterns can be stitched together to create a raw garment mesh. Then, we can compute symmetric axis of the raw garment mesh. The raw garment mesh can manually aligned with the body mesh based on their symmetric axes. Then, we can group the patterns that form a loop. For example, if the patterns form a loop and they are near to an arm, they may warp around the arm. For the meshes that cover the torso, these meshes should belong to the front body pattern or back body pattern. Then, these meshes should warp around the torso.

Computation of $M_G^{Initial}$. After we place the raw garment mesh and the body mesh M together, the raw garment mesh may intersect with mesh M or even it may intersect with itself. This is because the orientations of the meshes of patterns are incorrect (see Figure 8(a)). To handle the intersections between the raw garment mesh and body mesh M , we can apply the method proposed in [Volino and Magnenat-Thalmann 2006]. The method attempts to move the intersecting regions along the directions which minimize intersected contours. However, this method may not be appropriate to handle the self-intersected case (see Figure 8(a)). In this case, an appropriate way to correct the intersection problem is to reorient the meshes of patterns. For example in Figure 8(a), the left sleeve pattern warps around the arm and thus we can rotate the pattern around the arm axis to resolve the self-intersection problem. To speedup the process of computing the initial garment mesh $M_G^{Initial}$, we

can build a database to record this kind of special cases. At the runtime stage, we can retrieve the data from the database and determine the appropriate way (e.g., by affine transformations) to resolve the intersection problems. Once resolving all the intersections, we obtain the initial garment mesh $M_G^{Initial}$.

Draping Simulation. There are many physics representation models for garments. We can choose a model that has a fast convergence speed, such as position based dynamics [Müller et al. 2007], spring-mass system [Liu et al. 2013] and projective dynamics [Bouaziz et al. 2014]. Besides choosing appropriate the physics representation model for the garments, we also need to worry about the collision detection process. During the draping simulation, there would be collisions occurring between body mesh M and $M_G^{Initial}$. It would be costly to perform collision detection if the complexities of M and $M_G^{Initial}$ are high. To speedup the process, we can use coarse meshes of the body and $M_G^{Initial}$ to perform the draping simulation.

Another possible direction is that we can use bounding spheres and only perform vertex-sphere or sphere-sphere tests [Ye et al. Apr., 2014]. Figure 9(a) and (b) shows simulation results of a deformable surface without self-collision detection and with collision detection. If the accuracy is not a major factor, adopting sphere-sphere tests is a good alternative. The advantages of using moving sphere-sphere over the traditional method (e.g., vertex-triangle and edge-edge pairs) are two folds: 1) the computation cost is cheaper and 2) it is more robust for rounding off errors and numerical errors. There is no degenerate cases (e.g., parallel edges) for using sphere-sphere tests.

The body mesh M can be approximated by using spheres and the vertices of garment mesh are represented as spheres. The collision test for two moving spheres with linear velocities takes two steps: 1) solving roots of a quadratic equation; and 2) distance computation at the roots in an chronological order. If the distance of two spheres is smaller than or equal to a predefined threshold, the two spheres collide. Of course, if two spheres are moving away from each other, we can ignore the collision event. Figure 9(c) shows the spheres at the vertices of the deformable surface. The radiuses of spheres should be large enough for avoiding the ‘tunnel’ effect.

Summary of the Entire Process. We summarize the entire process in Algorithm 1. There are two output: the 3D garment mesh, M_G , and the stitching correspondences between patterns, C .

Algorithm 1 From pattern sewing to draping process

- 1: **Input:** Human body B , and a set of patterns T
 - 2: **Output:** garment mesh M_G , stitching result C
 - 3: Initialization
 - 4: Compute the edge stitching correspondences, C
 - 5: Manually fill out missing stitching data and update C
 - 6: Align patterns T and body mesh M
 - 7: Compute the initial garment mesh $M_G^{Initial}$
 - 8: Perform draping simulation for $M_G^{Initial}$ to obtain M_G
 - 9: Return M_G, C
-

Further, we can change the pose of body B to pose P . When necessary, another physics representation model can be switched. It would be interesting to interactively change the pose of the body mesh and to generate the draped garment on the fly.

14 Conclusions

A high degree of realism has been currently achieved in the design and interactivity of virtual humans and virtual agents in both on-line

Figure 9: a) Without self-collision detection; The deformable surface intersects with itself. b) With self-collision detection; c) Spheres at the vertices.

and off-line applications. Many of the hard problems, specifically due to level of detail and the sheer number of elements in their representation of their components, such as hair, skin, and clothing have been, to some extent, resolved with good approximations. This is readily observed in the current 3D game technology and CG-based cinematic effects. However, realistic clothing, woven and knitted materials, as well as multilayer composites are still presenting significant challenges both in their geometric representation as well as in their dynamic behavior in a variety of conditions. Some specific phenomena such as tear, burning and deformation changes subject to immersion into liquids or temperature changes are completely open problems. Much of the future work could potentially address these problems to a level of detail that should exceed the current solutions. But, one of the most important problems that remains very difficult to solve in pervasive applications is the relative compensation for the lack of an energy conservation principle in virtual environments.

15 Acknowledgments

We would like to express our gratitude to the Hong Kong RGC for the financial support of this work in the form of grants GRF PolyU 5101/11E, PolyU 5100/12E, PolyU 5100/13E, the National Science Council of ROC (Taiwan) under the grant no. NSC 102-2221-E-009-103-MY2 and the Ministry of Science and Technology of ROC (Taiwan) under the grant no. MOST 103-2221-E-009-122-MY3.

References

BAGON, S., BROSTOVSKI, O., GALUN, M., AND IRANI, M. 2010. Detecting and sketching the common. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 33–40.

BERTHOUSOZ, F., GARG, A., KAUFMAN, D. M., GRINSPUN, E., AND AGRAWALA, M. 2013. Parsing sewing patterns into 3d

garments. *ACM Trans. Graph.* 32, 4, 85:1–85:12.

BOUAZIZ, S., MARTIN, S., LIU, T., KAVAN, L., AND PAULY, M. 2014. Projective dynamics: Fusing constraint projections for fast simulation. *ACM Trans. Graph.* 33, 154:1–154:11.

BRIDSON, R., FEDKIW, R., AND ANDERSON, J. 2002. Robust treatment of collisions, contact and friction for cloth animation. *ACM Transactions on Graphics* 21, 3, 594–603.

BROCHU, T., EDWARDS, E., AND BRIDSON, R. 2012. Efficient geometrically exact continuous collision detection. *ACM Transactions on Graphics* 31, 4, 96:1–96:7.

CAI, Y., AND BACIU, G. 2011. Detection of repetitive patterns in near regular texture images. In *IVMSP Workshop, 2011 IEEE 10th*, 60–65.

CURTIS, S., TAMSTORF, R., AND MANOCHA, D. 2008. Fast collision detection for deformable models using representative-triangles. In *Proceedings of the 2008 Symposium on Interactive 3D Graphics and Games*, 61–69.

GUAN, P., REISS, L., HIRSHBERG, D., WEISS, A., AND BLACK, M. J. 2012. Drape: Dressing any person. *ACM Trans. on Graphics (Proc. SIGGRAPH)* 31, 4 (July), 35:1–35:10.

HAHN, F., THOMASZEWSKI, B., COROS, S., SUMNER, R. W., COLE, F., MEYER, M., DEROSE, T., AND GROSS, M. 2014. Subspace clothing simulation using adaptive bases. *ACM Trans. Graph.* 33, 4, 105:1–105:9.

HAYS, J., LEORDEANU, M., EFROS, A. A., AND LIU, Y. 2006. Discovering texture regularity as a higher-order correspondence problem. In *Computer Vision V ECCV 2006*, A. Leonardis, H. Bischof, and A. Pinz, Eds., vol. 3952 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 522–535.

HUANG, S., CAI, C., AND ZHANG, Y. 2009. Wood image retrieval using sift descriptor. In *Computational Intelligence and Software Engineering, 2009. CiSE 2009. International Conference on*, 1–4.

HUTTER, M., AND FUHRMANN, A. 2007. Optimized Continuous Collision Detection for Deformable Triangle Meshes. In *Proceedings of the 2007 WSCG*, 25–32.

IRANI, O., AND BOIMAN, M. 2007. Similarity by composition. In *Advances in Neural Information Processing Systems 19: Proceedings of the 2006 Conference (NIPS)*, vol. 19, MIT Press, 177.

JAIN, A., AND FARROKHNI, F. 1990. Unsupervised texture segmentation using gabor filters. In *Systems, Man and Cybernetics, 1990. Conference Proceedings., IEEE International Conference on*, 14–19.

JULESZ, B., AND BERGEN, J. 1983. Human factors and behavioral science: Textons, the fundamental elements in preattentive vision and perception of textures. *Bell System Technical Journal*, The 62, 6 (July), 1619–1645.

JULESZ, B. 1981. Textons, the elements of texture perception, and their interactions. *Nature* 290, 5802, 91–97.

JULESZ, B. 1984. A brief outline of the texton theory of human vision. *Trends in Neurosciences* 7, 2, 41–45.

JURIE, F., AND TRIGGS, B. 2005. Creating efficient codebooks for visual recognition. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, vol. 1, 604–610 Vol. 1.

- KANG, M. K., AND LEE, J. 2007. Technical section: A real-time cloth draping simulation algorithm using conjugate harmonic functions. *Computers and Graphics* 31, 2, 271–279.
- KAVAN, L., GERSZEWSKI, D., BARGTEIL, A. W., AND SLOAN, P.-P. 2011. Physics-inspired upsampling for cloth simulation in games. In *ACM SIGGRAPH 2011 Papers*, 93:1–93:10.
- KIM, D., AND YOON, S., 2014. OpenCCD: Continuous Collision Detection API, used in HPCCD. <http://sglab.kaist.ac.kr/OpenCCD>.
- KIM, D., HEO, J., HUH, J., KIM, J., AND YOON, S. 2009. HPCCD: Hybrid parallel continuous collision detection using CPUs and GPUs. *Computer Graphics Forum* 28, 7, 1791–1800.
- KLOSOWSKI, J., HELD, M., MITCHELL, J., SOWIZRAL, H., AND ZIKAN, K. 1998. Efficient collision detection using bounding volume hierarchies of k-DOPs. *IEEE Transactions on Visualization and Computer Graphics* 4, 1, 21–36.
- LAUTERBACH, C., MO, Q., AND MANOCHA, D. 2010. gproximity: Hierarchical gpu-based operations for collision and distance queries. *Computer Graphics Forum* 29, 2, 419–428.
- LEE, A., MORETON, H., AND HOPPE, H. 2000. Displaced subdivision surfaces. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, SIGGRAPH '00, 85–94.
- LIU, T., BARGTEIL, A. W., O'BRIEN, J. F., AND KAVAN, L. 2013. Fast simulation of mass-spring systems. *ACM Trans. Graph.* 32, 6, 214:1–214:7.
- MAGNENAT-THALMANN, N., AND THALMANN, D. 2007. *Handbook of virtual humans*. John Wiley & Sons.
- MALIK, J., AND PERONA, P. 1990. Preattentive texture discrimination with early vision mechanisms. *J. Opt. Soc. Am. A* 7, 5 (May), 923–932.
- MALIK, J., BELONGIE, S., SHI, J., AND LEUNG, T. 1999. Textons, contours and regions: cue integration in image segmentation. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 2, 918–925 vol.2.
- MEZGER, J., KIMMERLE, S., AND ETZMUSS, O. 2003. Hierarchical techniques in collision detection for cloth animation. *Journal of WSCG* 11, 2, 322–329.
- MÜLLER, M., HEIDELBERGER, B., HENNIX, M., AND RATCLIFF, J. 2007. Position based dynamics. *J. Vis. Comun. Image Represent.* 18, 2, 109–118.
- MUNDY, J., AND ZISSERMAN, A. 1994. Repeated structures: Image correspondence constraints and 3d structure recovery. In *Applications of Invariance in Computer Vision*, J. Mundy, A. Zisserman, and D. Forsyth, Eds., vol. 825 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 89–106.
- PABST, S., KOCH, A., AND STRÄßER, W. 2010. Fast and Scalable CPU/GPU Collision Detection for Rigid and Deformable Surfaces. 1605–1612.
- PROVOT, X. 1997. Collision and self-collision handling in cloth model dedicated to design garments. In *Graphics Interface*, 177–189.
- TANG, M., CURTIS, S., YOON, S., AND MANOCHA, D. 2009. ICCD: Interactive continuous collision detection between deformable models using connectivity-based culling. *IEEE Transactions on Visualization and Computer Graphics* 15, 4, 544–557.
- TANG, M., MANOCHA, D., AND TONG, R. 2009. Multi-core collision detection between deformable models. In *SIAM/ACM Joint Conference on Geometric and Physical Modeling*, 355–360.
- TANG, M., MANOCHA, D., AND TONG, R. 2010. Fast continuous collision detection using deforming non-penetration filters. In *Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 7–13.
- TANG, C., LI, S., AND WANG, G. 2011. Fast continuous collision detection using parallel filter in subspace. In *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 71–80.
- TANG, M., MANOCHA, D., LIN, J., AND TONG, R. 2011. Collision-streams: Fast GPU-based collision detection for deformable models. In *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 63–70.
- TANG, M., TONG, R., NARAIN, R., MENG, C., AND MANOCHA, D. 2013. A gpu-based streaming algorithm for high-resolution cloth simulation. *Computer Graphics Forum* 32, 7, 21–30.
- TERZOPOULOS, D., WITKIN, A., AND KASS, M. 1988. Symmetry-seeking models and 3d object reconstruction. *International Journal of Computer Vision* 1, 3, 211–221.
- TESCHNER, M., KIMMERLE, S., HEIDELBERGER, B., ZACHMANN, G., RAGHUPATHI, L., FUHRMANN, A., CANI, M.-P., FAURE, F., MAGNENAT-THALMANN, N., STRASSER, W., AND VOLINO, P. 2005. Collision detection for deformable objects. In *Computer Graphics Forum*, 61–81.
- TUYTELAARS, T., TURINA, A., AND VAN GOOL, L. 2003. Non-combinatorial detection of regular repetitions under perspective skew. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 25, 4 (April), 418–432.
- UMETANI, N., KAUFMAN, D. M., IGARASHI, T., AND GRINSPUN, E. 2011. Sensitive couture for interactive garment modeling and editing. *ACM Trans. Graph.* 30, 4, 90:1–90:12.
- UNC GAMMA GROUP. SelfCCD: Continuous collision detection for deforming objects. <http://gamma.cs.unc.edu/SELFCD>.
- VAN DEN BERGEN, G. 1999. Efficient Collision Detection of Complex Deformable Models using AABB Trees. *Journal of Graphics, GPU, and Game tools* 2, 4, 1–14.
- VOLINO, P., AND MAGNENAT-THALMANN, N. 2006. Resolving surface collisions through intersection contour minimization. In *ACM SIGGRAPH 2006 Papers*, 1154–1159.
- VOLINO, P., AND MAGNENAT-THALMANN, N. 2007. Stop-and-go cloth draping. *Visual Computer* 23, 9, 669–677.
- VOORHEES, H., AND POGGIO, T. 1988. Computing texture boundaries from images. *Nature* 333, 6171, 364–367.
- WANG, H. 2014. Defending continuous collision detection against errors. *ACM Trans. Graph.* 33, 4, 122:1–122:10.
- WONG, S.-K., AND BACIU, G. 2005. Dynamic interaction between deformable surfaces and non-smooth objects. *IEEE Trans. on Visualization and Computer Graphics* 11, 3, 329–340.
- WONG, S.-K., AND BACIU, G. 2005. Gpu-based intrinsic collision detection for deformable surfaces. *Computer Animation and Virtual Worlds* 16, 3-4, 153–161.

- WONG, S.-K., AND BACIU, G. 2006. A randomized marking scheme for continuous collision detection in simulation of deformable surfaces. In *Proc. of the ACM Int'l Conf. on Virt. Reality Continuum and Its Applications*, 181–188.
- WONG, S.-K., AND BACIU, G. 2014. Continuous collision detection for deformable objects using permissible clusters. *The Visual Computer*, doi: 10.1007/s00371-014-0933-6.
- WONG, S.-K., AND CHENG, Y.-C. 2014. Continuous self-collision detection for deformable surfaces interacting with solid models. *Computer Graphics Forum*.
- WONG, S.-K., BACIU, G., LIU, C.-M., AND YEH, C.-C. 2010. Robust Continuous Collision Detection for Deformable Objects. In *ACM Symp. on Virtual Reality Software and Technology*, 55–62.
- WONG, S.-K., LIN, W.-C., HUNG, C.-H., HUANG, Y.-J., AND LIU, S.-Y. 2013. Radial view based culling for continuous self-collision detection of skeletal models. *ACM Transactions on Graphics* 32, 4, 114:1–10.
- WONG, S.-K., LIN, W.-C., WANG, Y.-S., HUNG, C.-H., AND HUANG, Y.-J. 2014. Dynamic radial view based culling for continuous self-collision detection. In *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 39–46.
- WONG, S.-K. 2005. High performance virtual clothing dynamics. *Ph.D. Thesis, Computer Science, HKUST*.
- WU, C., FRAHM, J.-M., AND POLLEFEYS, M. 2010. Detecting large repetitive structures with salient boundaries. In *Computer Vision VECCV 2010*, K. Daniilidis, P. Maragos, and N. Paragios, Eds., vol. 6312 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 142–155.
- WU, C., FRAHM, J.-M., AND POLLEFEYS, M. 2011. Repetition-based dense single-view reconstruction. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, 3113–3120.
- YANG, A. Y., HUANG, K., RAO, S., HONG, W., AND MA, Y. 2005. Symmetry-based 3-d reconstruction from perspective images. *Computer Vision and Image Understanding* 99, 2, 210–240.
- YE, M., WANG, H., DENG, N., YANG, X., AND YANG, R. Apr., 2014. Real-time human pose and shape estimation for virtual try-on using a single commodity depth camera. *IEEE Trans Vis Comput Graph*. doi: 10.1109/TVCG.2014.35.
- ZHANG, X., AND KIM, Y. 2012. Simple culling methods for continuous collision detection of deforming triangles. *IEEE Trans. on Visualization and Computer Graph*. 18, 7, 1146 – 1155.
- ZHANG, J., BACIU, G., ZHENG, D., LIANG, C., LI, G., AND HU, J. 2013. Idss: A novel representation for woven fabrics. *Visualization and Computer Graphics, IEEE Transactions on* 19, 3 (March), 420–432.
- ZHAO, P., AND QUAN, L. 2011. Translation symmetry detection in a fronto-parallel view. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, 1009–1016.
- ZHENG, D., BACIU, G., AND HU, J. 2010. Entropy-based fabric weave pattern indexing and classification. *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI)* 4, 4, 76–92.
- ZHENG, D., HAN, Y., BACIU, G., AND HU, J. 2012. Design through cognitive color theme: A new approach for fabric design. In *Cognitive Informatics Cognitive Computing (ICCI*CC), 2012 IEEE 11th International Conference on*, 346–355.