Université de Lille
Ecole Doctorale MADIS

Laboratoire CRIStAL (UMR 9189)
Groupe Image, Equipe 3DSAM

3D and 4D Human body surface comparison and deformation: From geometric invariants to Riemannian shape analysis

Comparaison et déformations de formes humaine 3D et 4D: Des invariants géometriques à l'analyse Riemannienne de formes

Presentée et soutenue publiquement par EMERY PIERSON

Thèse en Spécialité Informatique

Le 3 Mars 2023

Composition du jury

Alain Trouvé Professeur, ENS Paris-Saclay	Rapporteur
Florent Dupont Professeur, Université de Lyon	Rapporteur
Mohamed Daoudi Professeur, IMT Nord Europe	Directeur de la thèse
Juan Carlos Alvarez Paiva Professeur, Université de Lille	Co-Directeur de la thèse
Olivier Colot Professeur, Université de Lille	Président du jury
Stefanie Wuhrer Chargée de Recherche, INRIA Grenoble Rhône-Alpes	Examinatrice
Alice Barbara Tumpach Maîtresse de conférences, University of Vienna	Examinatrice
Klaus Hildebrandt Assistant Professor, Delft University of Technology	Examinateur
Martin Bauer Associate Professor, Florida State University	Invité
Sylvain Arguillère Chargé de Recherche CNRS, Université de Lille	Invité

Abstract

In the broad field of computer vision, Three-Dimensional (3D) Human shape and body understanding has always been a hot topic. This thesis focuses on the 3D human body and its motion, with new objectives and problems arising from the growing availability of human body data. As articulated objects, human bodies are naturally complicated, as they evolve in identity and pose. Moreover, the human body shape space is built by filtering out preserving transformations, namely rigid motions, and reparametrizations. Under these challenges, our thesis proposes a 2-fold approach for the human body. In the first part of the thesis, we focus on 3D human shape sequence retrieval. We leave the shape space and propose to compare pose and motion in descriptor spaces. We start by describing the human pose, using the intuition that a human pose is nearly characterized by its convex hull. This hypothesis allows us to introduce three transformation-invariant surface descriptors. Using these invariants reduces the comparison of 3D+time surface representations to the comparison of polygonal curves. Experimental results are promising, but slight modifications are needed to work on noisy data. We improve our motion comparison results in a second time by seeing the human body surfaces as varifolds in a Hilbert space, with an inner product derived from a positive definite kernel. The sequences of 3D shapes are represented by matrices derived from the Gram matrix of the motion, reducing the problem of comparison of two 3D sequences of humans to a comparison of matrices. In the second part of the thesis, we focus on building a framework for generating human body deformations. We equip our shape space with a family of elastic Riemannian metrics that not only are invariant under rigid motions and reparametrizations but also allow us to quantify variations between human bodies under pose and shape changes. We start by working on registered shapes and use a low-dimensional approach to compute the interpolation between human bodies (geodesic path), and statistical tools (eg. mean of human shape). The main limitation is that human shapes need to be pre-registered to a template. To tackle this issue, we extend the previous Riemannian framework to human body scans. We do so by using the varifold metric as a parameterization regularizer. Moreover, we represent human bodies with a low dimensional latent space representation, which is equipped with the pullback of the Riemannian metrics, simplifying the encountered optimization problems. The registration and interpolation problems are solved jointly, allowing the framework to operate directly on unregistered shapes. Experimental results show significant improvements with respect to previous solutions in terms of shape registration, interpolation, and extrapolation. We demonstrate the utility of the proposed frameworks in pose and shape retrieval of the human body, motion transfer, and random generation of body shape and pose.

Résumé

Parmi les nombreuses thématiques de la vision par ordinateur, l'étude du corps humain, plus précisément l'analyse de sa forme en trois dimensions, a toujours été un sujet de grand intérêt pour ses nombreuses applications. La thèse se concentre sur le corps humain en trois dimensions (3D), ainsi que sur son mouvement, avec les nouveaux défis scientifiques posés par la disponibilité grandissante de ce type de données. Les corps humains sont naturellement compliqués en tant que corps articulés, évoluant à la fois en identité et en changement de pose. De plus, l'espace de formes des corps humains se construit en filtrant les transformations que sont les mouvements rigides et les re-paramétrisations.La thèse propose une approche en deux temps pour la compréhension du corps humain, tout en s'attaquant aux challenges énoncés. Dans un premier temps, nous nous concentrons sur la recherche des séquences de corps humains 3D dans une base de données. Nous proposons de quitter l'espace de formes en comparant les poses et les mouvements dans des espaces de descripteurs. Nous commençons par la description de la pose du corps humain, en supposant que la pose du corps humain est caractérisée par son enveloppe convexe. Cette hypothèse nous permet de construire trois descripteurs invariants par transformations géométriques. Nous réduisons la comparaison de séquences 3D+t à une comparaison de courbes polygonales. Les résultats expérimentaux sont satisfaisants, mais un (léger) travail supplémentaire est nécessaire pour l'utilisation sur données bruitées. Nous améliorons nos résultats sur le mouvement dans une seconde approche où l'on représente les surfaces de corps humains par des varifolds dans un espace de Hilbert, en utilisant un produit scalaire qui découle d'un noyau défini positif. Les séquences de formes 3D sont représentées par des matrices dérivées de la matrice de Gram du mouvement, ce qui réduit la comparaison de séquences de corps humains 3D à une comparaison de matrices. Dans un second temps, nous nous concentrons sur la construction d'un cadre général pour la génération des déformations humaines. Nous équipons l'espace de formes d'une famille de métriques Riemanniennes élastiques qui sont invariants par mouvements rigides et re-paramétrisations, qui ont de plus aussi l'avantage de distinguer les évolutions en pose et en identité. Nous nous restreignons d'abord aux corps humains alignés, et nous proposons une approche de dimension réduite pour l'interpolation (chemin géodésique) entre corps humains, ainsi que pour des outils statistiques (moyenne de corps humains). La limitation de cette approche réside dans la nécessité de pré-aligner les corps humains avec un modèle donné. Pour résoudre ce problème, nous améliorons l'approche en l'étendant aux scans bruts de corps humains. Pour y parvenir, nous utilisons la métrique varifold comme une quantité régularisant l'espace des paramétrisations. De plus, nous représentons directement les corps humains dans un espace latent de petite dimension, équipé des images réciproques des familles de métriques Riemanniennes. Cela permet de simplifier la résolution des problèmes d'optimisation pour le calcul de géodésiques. L'alignement et l'interpolation de formes sont résolus conjointement, ce qui permet de travailler directement avec des formes non alignées. Les résultats expérimentaux démontrent une amélioration significative par rapport aux solutions existantes pour l'alignement, l'interpolation et l'extrapolation de formes. Nous démontrons finalement l'utilité de ces approches pour la recherche de pose et d'identités, le transfert de mouvement et la génération aléatoire d'identité et de pose de corps humains.

Remerciements

Il est évident que cette thèse n'aurait pu avoir lieu sans l'aide, l'accompagnement de nombreuses personnes.

Tout d'abord, mon directeur de thèse, Mohamed. C'est sa convivialité et sa considération pour les autres qui m'ont convaincu de le rejoindre. Sans lui, je n'aurai jamais pu découvrir des sujets aussi intéressants ni m'intéresser un tant soi peu à la géométrie. De plus, il a toujours été, et est toujours de bonne volonté pour accompagner ses doctorants, disponible pour des discussions. Je réalise aujourd'hui à quel point j'ai eu de la chance de travailler dans son équipe.

Evidemment, mon co-directeur de thèse, Juan Carlos. Est-ce qu'il existe des thèsards de Mohamed ne le connaissant pas ? Sa complicité avec Mohamed, ainsi que sa jovialité permettent de faire fleurir sans arrêt de nouvelles idées qui s'avèrent toujours utiles et pertinentes. Je tiens aussi à saluer son abnégation, sa présence et sa patience face à mon impatience, malgré les évènements tumultueux qu'il devait affronter par ailleurs.

Bien sûr, il y a tout mes co-auteurs. D'abord Martin, qui m'a accueilli pendant un mois dans son laboratoire et a pu me faire découvrir de nouveaux horizons. Emmanuel évidemment, qui s'est pratiquement occupé de moi pendant mon séjour en Floride, et avec qui j'ai passé des soirées inoubliables. Une pensée aussi pour Barbara qu'il me tarde de finir par recontrer, Sylvain et Nicolas.

Merci aussi aux membres de mon jury, Florent Dupont, Alain Trouvé, Olivier Colot et Stefanie Wuhrer qui ont accepté de rapporter et d'examiner ma thèse. Ensuite mes collègues. Baptiste, Yujin et moi avons commencé nos thèses en même temps ou presque, et nous les finissons en même temps ou presque. Cela sera vraiment bizarre de me retrouver sans eux au moment d'aller manger, il est inestimable d'avoir pu avoir quelqu'un qui conaissait exactement la même situation que la mienne dans le labo. J'ai aussi une pensée similaire pour mes collègues lointains que je recontrais lors des réunions pour le projet ANR, Clément, Aymen, Pierre et Boyang, ainsi que leurs encadrants, Florent, Florence, Guillaume, Hyewon et Edmond. Un pensée aussi à Estèphe, Deise, Kévin, et récemment Thomas, qui n'ont rejoint plus récemment le contingent de l'équipe, et qui vont devoir se débrouiller sans nous (ou bien seront-ils content d'être débarassés de nous ?). Enfin, merci à ceux qui était là avant, Benjamin, Oussama et Naima. Je remercie aussi toute le département administratif du laboratoire qui rendent tout ce qu'on fait possible, en particulier Maryline, qui a dû s'arracher de nombreuses fois les cheveux pour gérer correctement les missions, et Dominique qui a toujours été coopérative.

Finalement, je me dois de remercier mes proches. Evidemment, tous mes amis qui ont

toujours été joviaux, et surtout compréhensifs, à propos de l'aventure dans laquelle je me lançais, Louise tout particulièrement. Je ne sais pas ce que tout cela aurait donné sans elle, elle qui m'a tenu par la main sans ressentiment pendant ces 3 ans. Il est fou de se dire qu'après ces presques 10 ans, on ait l'impression que tout cela ne fait que commencer. Parmi ceux qui méritent leur nom ici, je me risque à faire une liste : Antoine, Alexis, Alexis, Anaïs, Claire-Amélie, Julien, Julien, Mikael, Mélanie, Alexis, Pierre, Guillaume, Kévin, Corentin, Lisandre, de nombreuses personnes rencontrées à Supélec et qui méritent leur nom ici aussi, mais au risque d'en oublier, je leur dis qu'elles se reconnaîtront. Je remercie toute la famille de Louise aussi, Stéphanie, les Lefrère, Les Moreau et Lutchmanen (et feu notre ami Chipo !), qui m'a toujours acceuilli comme un membre à part entière, et tout particulièrement les Moreau, Constance et Camille en têtes, qui ont adouci chaque mardi, même quand ça n'allait pas. Finalement, je m'adresse à ma famille, Ira et Noah qui m'ont rejoint à Lille et pour qui je l'espère, la période des études sera aussi heureuse. Maman aussi évidemment, qui m'accompagne toujours même quand elle est loin, et toutes la famille Cannaké. Enfin, je souhaitais avoir une pensée à mon père, j'aurai aimé qu'il connaisse cette période de ma vie, et à la famille Pierson. Finalement, j'ai une pensée toute particulière pour Mamie Jacqueline, à qui j'aurai sincèrement aimé pouvoir expliquer ce que je faisais, et à laquelle je dédie ce manuscrit.

Contents

Notations

Shape space notations

S	$\triangleq$	A regular surface
$\mathscr{T}$	$\triangleq$	Human body template
G	$\triangleq$	Group of preserving transformations of a surface
$\mathrm{Diff}^+(\mathscr{T})$	$\triangleq$	Group of orientation preserving diffeomorphisms of the template
T	$\triangleq$	Translation
R	$\triangleq$	Rotation
$SO(3)$	$\triangleq$	Rotation group
$\mathscr{S}$	$\triangleq$	The surface space or pre-shape space
$\mathscr{H}$	$\triangleq$	Human shape space
$[f]$	$\triangleq$	Human shape
f	$\triangleq$	Template-parameterized surface, representative of $[f]$
$\bar{f}$	$\triangleq$	Template surface mapped to itself
h or k	$\triangleq$	Tangent vector of the surface space
$\big((\, , \,)\big)_f$	$\triangleq$	Riemannian metric on Tangent space T_f
$n, \vec{n}$ or $N(p)$	$\triangleq$	Normal of the surface (at point p)
g or g_p	$\triangleq$	First fundamental form of the surface (at point p)
Δ	$\triangleq$	Laplace-Beltrami operator
μ_S	$\triangleq$	Varifold corresponding to the surface S

Discrete mesh notations

M	$\triangleq$	A surface mesh
v_i	$\triangleq$	Vertex of index i
t_i	$\triangleq$	Triangular face of index i
n_i	$\triangleq$	Normal of t_i
c_i	$\triangleq$	Center of t_i

Chapter 1

Introduction

1.1 Thesis objectives

There has been an increasing interest in recent years in analyzing the shape and motion of the 3D human body. Advances in 3D human shape estimation algorithms, 3D scanning technology, hardware-accelerated 3D graphics, and related tools, are enabling access to large-scale 3D human body data. This data comes in the form of 3D surface meshes that may not correspond to coherent discretizations (different *surface parameterizations*). Moreover, the meshes may differ up to an irrelevant rigid motion. In this thesis, I propose a new geometrical approach that profits from the articulated nature of human bodies to study the 3D human body and human body motion under the enunciated constraints. More precisely, we will develop techniques for:
Geometry of the human body: When working on human bodies and their motions (see Figure 1.1 for a concrete example), the *human pose*, or the articulation of the human body in space, and the *human shape*, or the human body's identity are the two main characteristics on which we focus. We wish our methods to be able to disentangle those characteristics depending on the underlying application.
3D Human Shape Sequence Retrieval: Given a sample human body motion, we would like to retrieve similar motions in a fixed dataset. This task is essential in the context of large-scale databases. Many works exist for 2D images or 3D skeletons, but only a few works address the problem on 3D human body motion. We propose motion and pose descriptors of human shapes, that are invariant under reparameterization and rigid motion. Those descriptors also show robustness to noise in data acquisition.
Human body deformation: We would like to obtain a framework that allows practical use in a wide range of applications, such as human pose interpolation, human motion extrapolation and transfer, or new pose generation. Moreover, the framework should be robust to unparameterized body scans. We propose a Riemannian approach that takes profit from the availability of large-scale human motion data, allowing for realistic deformations. The desired applications are easily defined with available tools of Riemannian geometry.

1.2 Context and motivations

The understanding of the human body shape has captivated many people throughout history. Indeed, one of the first representations of the human body, the Venus of Willendorf, dates

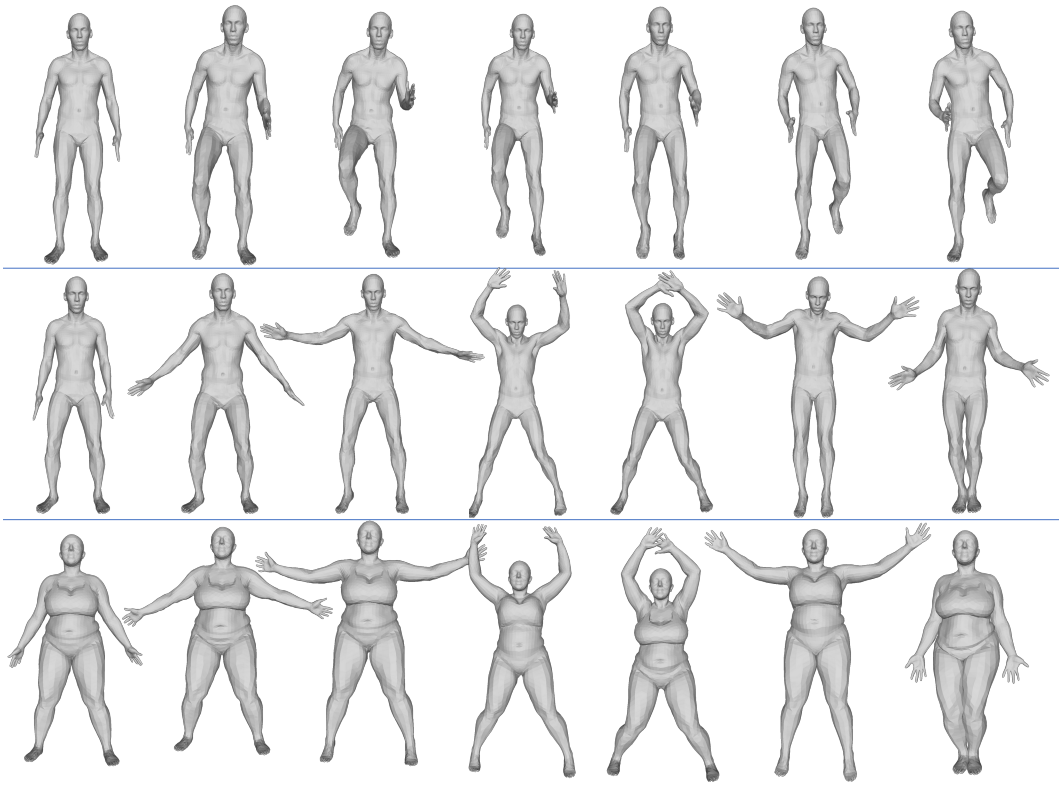


Figure 1.1 Three human motions: The first and second is executed by the same individual (same shape, different poses), the third is the second motion executed by a different individual (different shapes, same poses.

back to 25000 BC. This fascination has endured throughout history, with the famous drawing of Leonardo Da Vinci, or Bruce Neuman's recent Contrapposto studies (Gleyze, 2019) on human body motion. From a more scientific point of view, the corresponding research area has been active for a few decades. Research and development on creating 3D characters that can be animated dates back to the 1980s with contributions from the Pixar Studios (Lasseter, 1987), followed rapidly by other research works (Chadwick *et al.*, 1989; Scheepers *et al.*, 1997) where the authors were decomposing manually body parts, and rigging a skeleton to humanoid characters in order to animate them automatically. At the same time, 3D action recognition, using motion capture (a process which captures a set of sparse keypoints' location during a given motion) was also already a hot topic in the 1990s' (Campbell & Bobick, 1995; Gavrila, 1999), while learning of human characteristics was pioneered by some works such as eigenfaces (Turk & Pentland, 1991b) or 3D morphable models (Blanz & Vetter, 1999), that decompose human faces into several principal components for face recognition and deformation. With the emergence of deep learning, many 2D images related areas have known exciting breakthroughs, allowing computers to automatically recognize the action of several people given a video or motion capture data (Cao *et al.*, 2019). However while 2D data is showing its utility for many tasks, a picture/video of an individual remains the 2D projection of a specific view of its human body with possibly different lightning, scales, and other parts of the environment present. In the

same time, skeleton data is far from detailed information about the human body, as it contains only sparse information. Some human body related tasks thus remain impossible to solve using this type of data.

On the other hand, 3D human shapes analysis opens up many exciting possibilities in human understanding. Not only the view-invariant human motion analysis (Weinland *et al.*, 2006), but studying the morphing of the body during motion, ie. trying to be able to describe and reproduce how the human body deforms in between two poses (interpolation) or when given an initial velocity (extrapolation) has a variety of different applications. One could of course think about the virtual world avatars in an always more connected world. In the same time, medicals could benefit from it, where physiotherapists would love to be able to detect a degenerative execution of motion causing pain to patients, Figure 1.2. Close to this domain, in sports training, a player could progress better if he was able to see the details in the wrong execution of an essential gesture (in Figure 1.2, an example is shown with the tennis forehand).

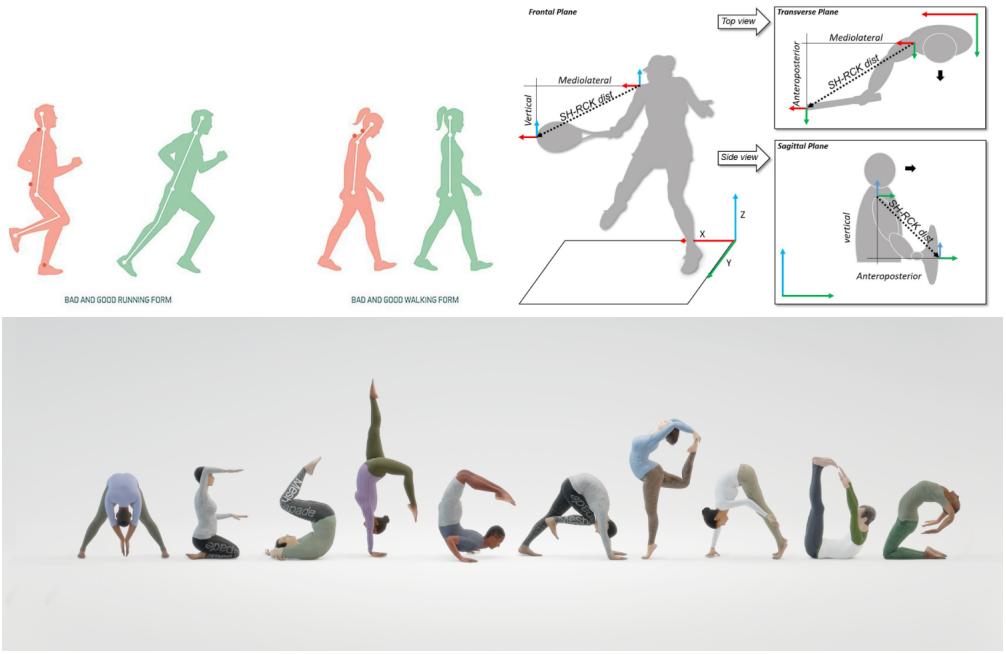


Figure 1.2 Different example of applications: Up-left: identifying bad and good postures form (https://calvarybible.org/dailyblog/2020/5/workout-wednesdays-may-20-2020). Up-right: Identifying the convenient Shoulder to racket distance in tennis forehand (Buszard *et al.*, 2020). Down: A few avatar examples from Meshcapade (https://meshcapade.com/home).

In the meantime, 3D human body data has become more and more available, with new setups that are less and less constrained for capture (for example, the Kinovis platform has a total capturable area of 100 m^2) and can be used to collect data related to the applications enounced before. Modern setups similar to those displayed in Figure 1.3 allow for detailed and scalable 3D human body deformation capture.

This evolution makes 3D human body data increasingly more available (Gkalelis *et al.*, 2009; Bogo *et al.*, 2014, 2017; Mahmood *et al.*, 2019; Cai *et al.*, 2022) than before. Combined with the promise of machine learning algorithms, several breakthroughs in this area have

Figure 1.3 Different type of 4D acquisition systems: 4D Scanner at MPI Tübingen, Kinovis multi-view 4D acquisition room, Shangai artificial laboratory 4D multi-view acquisition system

happened in the last year, with the widely used Skinned Multi-Person Linear (SMPL) body model (Loper *et al.*, 2015), or recent mesh autoencoders (Bouritsas *et al.*, 2019; Lemeunier *et al.*, 2022). However, 3D human body data remains of low availability compared to usual 2D images, because of the high recording cost compared to single-view camera recordings. Combined with the nature of the human body as a 3D surface, which suffers several geometric challenges (see next section) that are not (or are less) present in 2D images, the application of deep learning for 3D data is still inefficient in many tasks. In particular, the neural network will tend to overlearn characteristics from training data such as surface discretization, while well-defined shape descriptors are less powerful (and do not include any data-driven information), but often guarantee theoretical invariances. At the same time, retrieving SMPL body model (or other body models) information of a shape is a costly procedure that can require manual intervention. Therefore, we believe that the output of well-known geometric invariants can benefit the different proposed tasks of the thesis.

As stated above, the growing availability of 3D data is a challenge in itself: before synthesizing animation, one needs to be able to find easily corresponding motions, possibly without the need for annotations (second objective). We should in addition expect this process to be the least sensitive to possible noise during motion acquisitions. After these steps, one needs to be able to synthesize human motions in the most realistic way and have the possibility of using mathematical tools for computing shape statistics (third objective). Those two steps are almost impossible without taking into account the geometry of the human body (first objective).

1.3 Experimental setups: datasets, evaluation metrics

In this thesis, we conduct several experiments on the retrieval and deformation of shapes. We resume in this section the main datasets in use, along with the most common evaluation metrics.

1.3.1 Datasets

FAUST dataset

The FAUST dataset (Bogo *et al.*, 2014), originally designed for mesh registrations, consists of 3D scans of 10 subjects in 30 different poses and is divided into training and testing sets. In the training set the 3D surfaces are registered to the SMPL human body template. We use those registrations, which are available for 10 different poses, as a dataset for static human pose

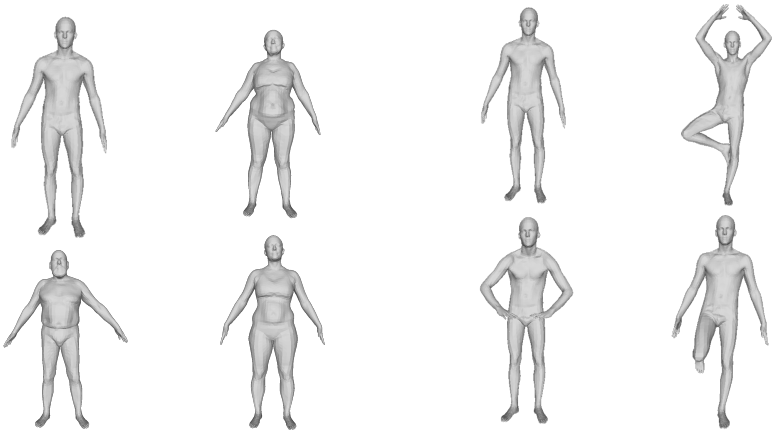


Figure 1.4 Left: A few identities of the FAUST dataset. Right: A few poses of the FAUST dataset.

retrieval and use a subset of the scans to evaluate the quality of deformed shapes in Chapter 7. Some samples are shown in Figure 1.4.

CVSSP3D datasets

The CVSSP3D datasets (Starck & Hilton, 2007; Gkalelis *et al.*, 2009) are 3D human motion datasets created for surface animation. It contains two parts: (1) a synthetic dataset, which contains artificial surfaces animated using known motion capture sequences, and (2) a real dataset, which contains the reconstruction of 3D human motions from multi-view video sequences. We summarize them as follows:

- *Synthetic dataset*. A synthetic model (1290 vertices and 2108 faces) is animated thanks to real motion skeleton data. Fourteen individuals performed each 28 different motions: sneak, walk (slow, fast, turn left/right, circle left/right, cool, cowboy, elderly, tired, macho, march, mickey, sexy, dainty), run (slow, fast, turn right/left, circle left/right), sprint, vogue, faint, rock n'roll, shoot. It has already been used (Huang *et al.*, 2010) for static shape evaluation in the context of 3D motion analysis. A motion from this dataset is presented in Figure 1.5, right. The sampling of the sequences is set to 25Hz.

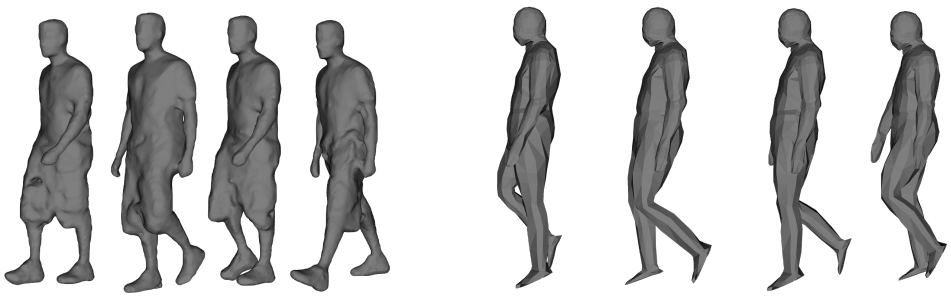


Figure 1.5 CVSSP3D dataset: walking motion from real data (left), fastwalk from synthetic data (right)

- *Real dataset*. This dataset contains 8 models performing 12 different motions: walk, run, jump, bend, hand wave (interaction between two models), jump in place, sit and stand up, run and fall, walk and sit, run then jump and walk, handshake (interaction between two models), pull. The number of vertices for each model vary around 35000. The sampling of the sequences is set to 25Hz. For the sake of simplicity, we follow usual practices (Veinidis *et al.*, 2019) and ditch out the two interaction actions in our experiments.

  As the reader can see in Figure 1.5, left, some of the motions in this dataset represent humans moving in loose-fitting clothes. The sensitivity of the reconstructed surface to clothes induces presence of noise in the meshes (see Figure 4.6) which makes it a challenge for 3D human motion retrieval.

D-FAUST dataset
The Dynamic FAUST (D-FAUST) (Bogo *et al.*, 2017) dataset contains high-quality scans, along with their corresponding registered meshes that will be used as training data. 10 individuals performed at most 14 different treadmill motions (hips, knees, light hopping stiff, light hopping loose, jiggle on toes, one leg loose, shake arms, chicken wings, punching, shake shoulders, shake hips, jumping jacks, one leg jump, running on spot), which means that the individual only moves along the height axis. For the registered meshes, a human template (6890 vertices) is registered to human body scans sampled at 60 Hz, and sequence length varies from 150 to 1200 (average 323). Due to the high speed of the recording, D-FAUST scans contain several

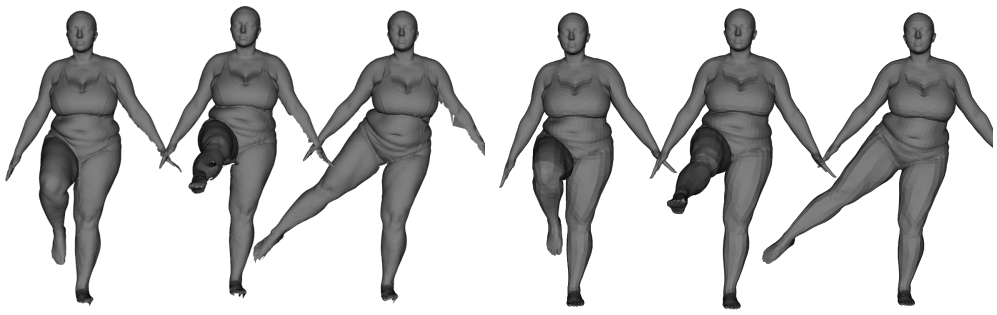


Figure 1.6 Knees motion from the D-FAUST dataset. Left: original scans, Right: registration available in Dyna dataset

singularities in the reconstructed surface, such as holes or even artificial objects (eg. parts of walls). Some of the registrations failed because of the complexity of D-FAUST pipeline, but a full set of registrations are available in Dyna dataset (Pons-Moll *et al.*, 2015).

Dyna dataset
The Dyna dataset contains all sequences registered to a human template (6890 vertices). We use those sequences in motion retrieval, where the meshes are sampled at 60 Hz, and sequence length varies from 150 to 1200 (average 323).

1.4 Thesis challenges

The thesis objectives stated in section 1 face several challenges, (1) that come with the use of 3D surfaces, and (2) that are specific to the study of the human body. The ones addressed in this thesis are summarized below.

Human body representation The human body motion is usually captured with a set of multi-view cameras and reconstructed from well-established reconstruction algorithms. However, while those algorithms are very efficient for the reconstruction of 3D surfaces from static scenes, handling moving 3D human shapes, in terms of pose (speed compared to cameras capture windows) and shape (dynamic shape motions, or noise induced by clothes) can be tedious. This can often induce noise in the reconstructed shapes (Figure 1.7). Moreover, the motion is preserved by rigid motions (*translation* and *rotation*) and there is no canonical robust way of finding those transformations afterward. Designing shape representations that are invariant to rigid motions and robust to noise will be a tough problem. In the meantime, the representation must be adapted to the context where it applies. In motion retrieval, computation time efficiency will be crucial. On the contrary, in human deformation synthesis, the quality of output shapes will be more important than computation efficiency.

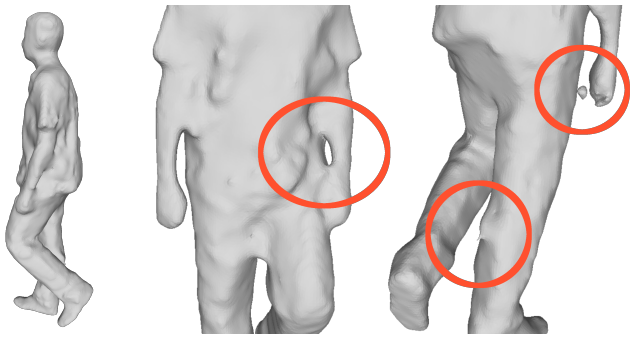


Figure 1.7 Noise in human shape acquisition. It can be of various forms, topological noise (middle), vertex noise (right bottom) or disconnected components (right top).

Human body parameterization The previous challenge is also linked to surface discretization. There is no way to ensure that the captured human body will have a canonical mesh, without requiring intensive registration algorithms which often require manual refinement. This leads us to a second set of preserving transformations, the surface parameterization. We need to find a way to represent the human body independently of the surface parameterization in order to be robust to human body scans.

Human body shape and pose disentanglement In this thesis, we work by separating human body shape and pose. While those notions are *almost* disentangled (two different individuals cannot execute *exactly* the same set of motions), they are well separated geometrically: the shape information is mainly contained in the intrinsic properties (local lengths, areas, and volumes) of the human surface, while the pose information is mostly contained in the extrinsic properties (joints coordinates in space) of the surface. We want to encode only one or separate that information (for example, keep only the pose information when comparing shapes for human motion retrieval, or transfer the pose evolution to a new identity when executing motion transfer).

Human body variability Finally, human surfaces are not any kind of surfaces. Some transformations cannot apply to them: for example, one cannot shrink or shear the hand of a human. We say that they are articulated surfaces, which means essentially that human shapes are almost rigid by parts, in other terms that the surface is built over the human skeleton and (mostly) changes through rotations around joints. This is formulated mathematically as follows: when one interpolates between two registered human poses, the resulting path is likely to leave the human body shape space (Figure 1.8). A human shape space candidate must propose a way to stay in the human body shape space when generating new poses or interpolation two poses.

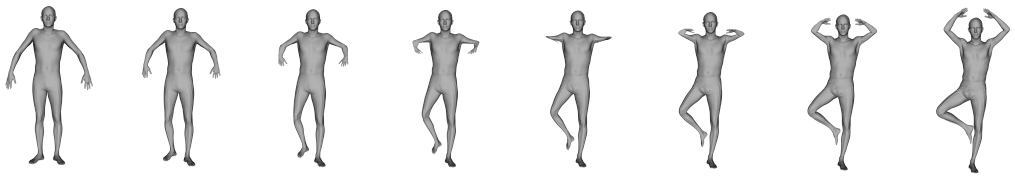


Figure 1.8 Linear interpolation between two human poses. The arms and the leg shrink during the interpolation, so the middle shapes are out of the human body shape space.

1.5 Thesis contributions

In this thesis, we bring four main contributions to human shape understanding, described below :

Invariant descriptor of the human pose - application to human motion : The first work is dedicated to the description of the human pose of the human body to classify specific poses. We propose the convexity hypothesis and say that the human pose can be approximated by the human convex hull, to build a human pose descriptor. This descriptor is then combined with a time series comparison algorithm (Dynamic Time Warping) to classify human motion.

Invariant descriptor of human motion : We quit the domain of human descriptors to build now directly human motion descriptors. We see human body surfaces as varifolds, which gives us a parameterization invariant representation, that can be embedded into an infinite dimensional Euclidean space using a Reproductive Kernel Hilbert Space (RKHS). We build a rotation and translation invariant motion descriptor using the inner product over the RKHS and the Gram matrix trick. The resulting descriptor allows us to compare human motion. The accuracy results improve compared to the previous contribution.

Riemannian human shape space for human deformation assessment : Starting from this point, we focus on the last thesis objective. We wish to provide a framework for human deformation. We define a human shape as an embedding from a human template to the 3D space and equip this space with a 3-parameter Riemannian metric. The parameters can be set to differentiate either shape or pose deformation. By incorporating human motion data in our pipeline, when interpolating two shapes, the resulting geodesic paths are visually realistic human deformations.

Elastic matching framework for unparameterized human deformation : We extend the previous chapter to unparameterized human shapes. We do so by using the varifold fidelity metric as a parameterization regularizer and use a similar but extended 6-parameter Riemannian

metric to match a human template to human shapes. We again take profit of available human body data and construct a latent space of human deformations, separated deformation in shape and pose directions. This allows us to properly parameterize the input human scans. We then demonstrate the capacities of the framework to several applications: interpolation, extrapolation, motion transfer, and random pose generation. We wish to emphasize that this work is made in collaboration with Nicolas Charon, Martin Bauer and its Ph.D. student Emmanuel Hartmann. Emmanuel and I contributed equally to the outputs of this Chapter.

1.6 Thesis outline

We retain the following organization for the manuscript. Chapter 2 presents related works in the context of 3D human shape understanding. In chapter 3, we introduce the mathematical concepts we use in the manuscript. Chapters 4,5,6 and 7 are dedicated to presenting the methodology behind each of the thesis contributions presented above. Finally, we conclude this manuscript and discuss what research perspectives it brings.

Publications

Publications and pre-prints that serve as basis for the manuscript :

P1 Emery Pierson, Juan Carlos Álvarez Paiva, Mohamed Daoudi. Projection-based classification of surfaces for 3D human mesh sequence retrieval. *Computer & Graphics, vol. 102, pp. 45-55, 2022* (*Oral presentation at Shape Modeling International Conference (SMI 2021)*

P2 Emery Pierson, Mohamed Daoudi, Sylvain Arguillère. 3D Shape Sequence of Human Comparison and Classification Using Current and Varifolds. *17th European Conference on Computer Vision (ECCV 2022), pp. 523-539*

P3 Emery Pierson, Mohamed Daoudi, Alice Barbara Tumpach. A Riemannian Framework for Analysis of Human Body Surface. *IEEE Winter Conference on Applications of Computer Vision (WACV 2022), pp. 2763-2772*

P4 Emmanuel Hartman*, Emery Pierson*, Martin Bauer, Nicolas Charon, Mohamed Daoudi. BaRe-ESA: A Riemannian Framework for Unregistered Human Body Shapes. *Arxiv pre-print, currently submitted to IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2023)* (* Emmanuel Hartman and I contributed equally to the paper.)

Other Publication:

- Emery Pierson*, Thomas Besnier*, Mohamed Daoudi, Sylvain Arguillère. Parameterization Robustness of 3D Auto-Encoders. *Eurographics Workshop on 3D Object Retrieval (3DOR 2022) Short Papers, pp 17-23* (* Thomas Besnier and I contributed equally to the paper)

Chapter 2

State of the art in 3D human body motion analysis

Contents

2.1 3D human shape descriptors

In order to describe properly one surface, the first approach proposed in the literature consists of designing discriminative, yet representative feature vectors, called shape descriptors. We will divide those descriptors into two different approaches: local descriptors mapping each point to a high dimensional vector (for tasks such as shape matching), and global descriptors that map the whole surface to a high dimensional feature vector and can be used in classification or retrieval.

2.1.1 Local descriptors

When describing a 3D surface, one of the first ideas that come across is to look at its geometric invariants. We know, from the fundamental theorem of surface that knowing the map from a surface to its normals and local metrics (or local distances) is identical to knowing the surface up to translation. One can thus think about taking the normal, but this quantity is not rotation invariant and is not discriminative enough for matching. In a mesh, local distances are often the length of edges, which is not invariant to parameterization. A good descriptor is then the derivative of those quantities, the curvature, given as the set of Mean and Gaussian curvatures or as principal curvatures (Faugeras & Hebert, 1986). However, their computation can be tedious in the discretized case. To overcome this, the shape index (Koenderink & Van Doorn, 1992) maps the point to a value between -1 and 1 encoding the local shape of the surface, ensuring robustness against scale changes.

While the curvature values range expresses faithfully what a shape looks like locally, it remains hard to describe more of the details that can appear in a human shape: for example, the middle of a finger has the same shape index as the forearm. To overcome this problem, point features, which are based on aggregating multiple quantities around each point of the surface have been an active area. After the pioneering work of Spin Images (Johnson, 1997), which takes local "histogram" pictures of the image from a chosen frame, multiple works have followed such as point signatures (Chua & Jarvis, 1997), shape context (Belongie *et al.*, 2000) or multi-scale curvature approaches (Li & Guskov, 2005).

Another family of approaches uses the Laplace-Beltrami operator on meshes, defined using the cotangent formula (Crane, 2018), generalizing its continuous counterpart. This operator is popular because it is able to capture efficiently the intrinsic properties of the surface. Descriptors are derived as solutions from specific surface equations that involve the Laplace operator. They are generally solved using the Eigendecomposition of the operator and by decomposing the solutions in the Eigenvector basis, allowing fast implementations. Among others, the Heat Kernel Signature (HKS) (Sun *et al.*, 2009; Bronstein & Kokkinos, 2010) derives from the heat equation and the Wake Kernel Signature (WKS) (Aubry *et al.*, 2011) from the Schrödinger wave equation. This approach has the advantage to be invariant to remeshing and to some extent, to isometries, aka transformations that do not change intrinsic distances. It was used successfully to match objects in shape-matching frameworks (Bronstein *et al.*, 2011) or later to capture identity information of the human body.

In the context of the human body, a recent study compares the abilities of different isometry invariant descriptors for human shapes. The Area-Projection Transform (Giachetti & Lovato, 2012), which measures locally how a point is likely to be part of a local cylinder or sphere

of size R, was the most successful for this task. While those descriptors are successful in shape matching and can be combined with registration algorithms such as Iterative Closest Point (Arun *et al.*, 1987) or functional maps (Ovsjanikov *et al.*, 2012) for shape registrations, a lot of those local descriptors contain intrinsic information. They are likely to fail for describing mostly extrinsic quantities such as the human pose. More global feature approaches are in general more efficient.

2.1.2 Global descriptors

While capturing local curvature gives information on the local shape, having information on the whole normals' distribution gives information on the local orientation of the shape, and so on the way it is embedded in space. Extended Gaussian Image (EGI) (Horn, 1984), and its complex version (CEGI) (Kang & Ikeuchi, 1991) are two spherical signals representing the distribution of normal directions (spherical points) on the shape. While those descriptors ensure theoretical guarantees on convex shapes (Gardner, R. J., 2006; Sellaroli, 2017), their expressivity takes a hit on general objects that are piecewise stack of convex shapes. Vranic (Vranic, 2004) proposed in his thesis a whole family of descriptors build as spherical signals. This work was extended in (Papadakis *et al.*, 2008), where the authors propose a spherical histogram descriptor, that became popular for human body pose and motion description (Veinidis *et al.*, 2017, 2019). Those approaches can be seen as part of the work in 3D space histograms, which was first proposed on regular grids (Ankerst *et al.*, 1999; Mian *et al.*, 2006) but suffers from translation and rotation challenges. The spherical Fourier analysis applied to well-crafted spherical signals allows easy building of invariants from space histograms on regular spherical bins (Kazhdan *et al.*, 2003; Tombari *et al.*, 2010). Rather than gathering information on 3D space location, Shape Distributions (Osada *et al.*, 2002) works on the set of pairwise distances (euclidean or geodesic (Hamza & Krim, 2003)). The histogram of those distances is a powerful descriptor that is naturally invariant to rotation and translation, only influenced by a scale factor.

Topological analysis of 3D meshes allows building descriptors on top of skeletal shapes, by using Reeb graphs to retrieve the skeleton corresponding to a mesh (Tierny *et al.*, 2006, 2007). It was later successfully applied to the human body, by computing the distance between human shapes as the distance between curves of the topological skeleton, called Extremal Human Curves (Slama *et al.*, 2013) and efficient for differentiating changes in pose and motion (Slama *et al.*, 2014). In the domain of intrinsic descriptors, the Laplace-Beltrami spectrum (Kac, 1966), which is the vector of eigenvalues of the Laplace-Beltrami operator, can be used as a *ShapeDNA* (Reuter *et al.*, 2006) for surfaces.

With the arrival of powerful local descriptors, multiple aggregation strategies in order to build up global descriptors from them have come. Histograms of descriptors are popular, with numerous tricks to avoid computational cost (Rusu *et al.*, 2009; Stein *et al.*, 1992; Rusu *et al.*, 2010). Bag-of-words strategies can also be used. The information of local descriptors is gathered into decomposable key points in the descriptor space (the *words*). A global descriptor of the shape is built by "counting" the number of instances of each of those words. Numerous bags of features were proposed from local descriptors, such as HKS (Bronstein *et al.*, 2011), WKS or Area Projection Transform (Giachetti & Lovato, 2012). Those approaches were confronted against each other in the SHREC 2016 challenges (Pickup *et al.*, 2016) for human shape retrieval.

A lot more descriptors exist in the literature (Biasotti *et al.*, 2016), but it is hard to find a suitable descriptor for human shapes, because of two reasons. A lot of them are too general and don't take care of human shape specificities, and on the other hand, they all have a different kind of invariance against preserving transformations. They can become approximately invariant to rotation using Principal Component Analysis (PCA) on vertices, or Continuous-PCA (Vranic, 2004), but doing so is sensitive to noise or significant change in pose. A successful approach would likely avoid it by building invariant descriptors.

2.2 Animation of 3D Humans

2.2.1 Human shape morphing

Human (or humanoid) shape morphing is a long time problem in computer graphics, as stated in the introduction. Numerous approaches were proposed to solve it. They propose new spaces of coordinates for shapes or build deformation energies to be optimized directly on the shape space.

Interpolating between two human shapes will undeniably induce unnatural deformation of the human body, thus leaving the human shape space. Mapping the 3D coordinate of human shapes to a more efficient space can be a way to solve this problem. Some well define coordinates can indeed live in a well-behaving space, where the interpolation produces plausible human deformations. This is the case of Linear Rotation invariant coordinates (Lipman *et al.*, 2005), which build local features from the surface local coordinate frames. Another approach consists in using the derivatives of those coordinates (Alexa, 2003) from the Laplace Beltrami operator and aggregating them on each vertex (from the vertex's neighborhood). The Lie Bodies (Freifeld & Black, 2012) approach consists in representing triangles as transformations from a canonical triangle: the shape space is mapped to the Lie group of transformations (rotation, translation, shear) of each triangle. This approach has been later improved with a more complete set of transformations and optimized for large deformation computation (Bansal & Tatu, 2018). The discrete fundamental theorem of surfaces states that lengths and dihedral angles define a mesh up to rotation and translation (first and second fundamental forms). This space of lengths and angles is used for defining the space of Discrete shells (Grinspun *et al.*, 2003), Isometry invariant coordinates (Baek *et al.*, 2015) or Non-Linear Rotation invariant Coordinates (Sassen *et al.*, 2020) (NRIC). In a similar approach, Von-Tycowicz *et al.* (Von-Tycowicz *et al.*, 2015) defined the strain tensor, as a local deviation of the local Riemannian metric of the surface from the identity tensor. They measure the deformation using the strain tensor's Eigenvalues. Coordinate-based deformation can also be used to define *part-by-part* deformation, by modifying specific landmarks of a shape. The latter is generally propagated to the whole body shape by Laplacian propagation (Sorkine *et al.*, 2004), solving Poisson equation (Xu *et al.*, 2005), or doing it As-Rigid-As-Possible (Sorkine & Alexa, 2007).

Coordinate spaces are often prone to error in the case of large deformation. Another possible approach is to define the deformation of shapes as the solution to an optimization problem. The challenge is thus to define a deformation energy that would be minimal when the deformation (or the path of deformations) is visually plausible. Those energies can be defined directly on the shape and be As-Rigid-As-Possible (Alexa *et al.*, 2000) (ARAP), As-Isometric-As-Possible (Kilian *et al.*, 2007) or take inspiration from physical observation like

the shell energy (Terzopoulos *et al.*, 1987). This energy, decomposed in membrane and flexural energies in discrete shells (Grinspun *et al.*, 2003), can be applied to some of the previously presented coordinate systems, and even be seen as a Riemannian metric on the NRIC manifold (Sassen *et al.*, 2020). Most of those optimized approaches rely on well-defined but non-convex energies, i.e. the optimization process is not trivial. In order to overcome this, multiscale approaches reduce first the dimension of the problem to find a good initialization for the full problem (Winkler *et al.*, 2010; Chu & Lee, 2009). They thus allow improvement of the optimization pipeline and the reduction of computation costs (Zhang *et al.*, 2015). Moreover, data-driven approaches can be proposed to define low dimensional subspaces of deformation (Baran *et al.*, 2009; Fröhlich & Botsch, 2011; Heeren *et al.*, 2018), and suits well to the case of human deformation where training data can be easily available.

While those approaches have very useful applications in computer graphics, they suffer from two drawbacks: the distances they define are not practical in the sense that they do not differentiate shape and pose changes, in other terms they are not suitable for human shape analysis, secondly they cannot be used in the unparameterized setting. A registration method is needed before applying them if one wants to interpolate between two unparameterized surfaces like human body scans. Some methods, like functional maps (Ovsjanikov *et al.*, 2012) can introduce significant bias in registration that later influences the statistical analysis, and other ones require supplemental information such as motion capture markers locations (Loper *et al.*, 2014).

2.2.2 Human shape spaces

Learning human shape spaces can solve the problem of pose and shape disentanglement, by specifically defining the space as the sum of well-defined human shape and pose spaces.

In order to model the pose space, most approaches follow the forward skinning approach (Lewis *et al.*, 2000), which consists to rig a skeleton to each point of the mesh. The weight attributed between a point and a joint represents how much a point *belongs* to a joint. Each joint rotation is applied to a mesh point accordingly to the skinning weights. For the identity space, most of the works are inspired by the Eigenfaces (Turk & Pentland, 1991a) and Morphable Models (Blanz & Vetter, 1999) pioneering works on faces. The approach consists of applying the idea of principal components on human shapes, allowing us to describe the human shape space and to generate new identities (Allen *et al.*, 2003).

The SCAPE model (Anguelov *et al.*, 2005) proposed a unification of those two approaches by learning the weights of the skeleton skinning method and following the previous works by modeling the identity space as a PCA space.

Several improvements have followed over the years based on these seminal works. Hasler *et al.* (Hasler *et al.*, 2009) propose a lightweight template mesh (around 1500 vertices) and a muscledeness function to increase the realism of their human body model. The arrival of wider datasets allows sanitizing of the preprocessing steps, which were including a lot of manual intervention, resulting in better registrations (Pishchulin *et al.*, 2017). This resulted in much more expressive and interpretable models in terms of identities.

Nowadays, the Skinned Multi-Person Linear (SMPL) model (Loper *et al.*, 2015) is the most popular model among others ones. It is widely used for human body reconstruction from images, and the release of the AMASS dataset (Mahmood *et al.*, 2019) in the SMPL parameter

format allowed its application on a high set of real animations (collected using MoCap data) to numerous human identities. Its derivation by adding hands (Romero *et al.*, 2017), feet (Osman *et al.*, 2022), face (Pavlakos *et al.*, 2019) and dynamic models (Bogo *et al.*, 2017) allows to improve the realism of human bodies. Recent iterations of this approach, based on more and more data availability, allowed results in a more expressive body shape space while having less demanding resources (Osman *et al.*, 2020, 2022). Recently, the arrival of deep learning generative models showed promising results towards replacing the PCA human shape space with more expressive models (Xu *et al.*, 2020).

In recent years, recent approaches propose to rig a skeleton to implicit function (Alldieck *et al.*, 2021; Deng *et al.*, 2020; Chen *et al.*, 2021) (signed distance field or occupancy networks, represented by neural networks) representing the surface, or to a neural radiance field (Xu *et al.*, 2021; Noguchi *et al.*, 2021). It permits a usable representation of the human body while being independent of any template mesh. Those approaches show promising results, but SMPL and its derivatives remain the most popular and easy to use because of their compatibility with computer graphics algorithms.

Those approaches show great expressivity and are easy to use, however, the retrieval of human body parameters for a given unparameterized scan is nowadays a costly and/or inexact procedure. Moreover, the use of a predefined skeleton allows us to define easily human body transformation but forces an arbitrary prior to human bodies. Those models are still prone to errors when generalizing to any kind of body. The growing availability of human body data today opens up the possibility to generalize other type of learning approaches to this problem.

2.3 Elastic matching

The field of elastic matching is a set of unparameterized approaches, which means that those approaches allow comparing and deforming shapes that do not share the same mesh or parameterization. The idea is to work on the surface space to find the shape space geodesics. One first defines the transformations (listed in the introduction) that preserve the shapes of surfaces. Then, the surface space is quotiented by the transformation space, which is done by solving optimization problems over the transformation space. The resulting shape space can be equipped with a Riemannian metric which is the "projection" (we will see what it means in detail in Chapter 3) of a transformation invariant Riemannian metric on the surface space (which means that the value of the metric doesn't change if we apply a different parameterization to the surfaces). The advantage of proceeding this way is that working on Riemannian manifolds allows to decline classical statistical tools for shape analysis: mean, Tangent PCA, ... Multiple metrics have been used in this kind of approach. The Square Root Normal Field (SRNF) representation (Jermyn *et al.*, 2012), is one of the preferred approaches because it simplifies greatly the computations, thanks to the fact that the corresponding distance has a closed form solution. The approach is very general and can be applied to numerous problems (Laga *et al.*, 2022; Jermyn *et al.*, 2017; Kurtek *et al.*, 2013) and is compatible with data-driven approaches. In particular, it was successfully applied to human shapes in (Laga *et al.*, 2017, 2022). The SRNF metric is part of the family of Dewitt metrics (Ebin, 1970; DeWitt, 1967), which are well suited for human shape because it can be parameterized to approach an as-isometric-as-possible setting (Kurtek *et al.*, 2012). The family of metrics was later simplified in (Su *et al.*, 2020b),

with closed-form formula of the distance for the full metric. Other metrics were proposed, like the Sobolev metrics (Bauer *et al.*, 2011; Su *et al.*, 2020a), curvature-weighted metrics *et al.* (Bauer *et al.*, 2012), or the Gauge invariant derivation of the Dewitt metrics (Tumpach *et al.*, 2016). The biggest drawback of those approaches is that they see human shapes as closed genus-0 surfaces, thus as a map from the sphere to the 3D space, which involves solving the difficult spherical parameterization problem (Praun & Hoppe, 2003). They then represent the space of reparameterizations as the reparameterizations of the sphere, which is a rather easier space to work on than the remeshing space. However, by mapping the surface to the sphere, one can lose information on high curvature spots of the surface, which results in the loss of information contained in original scans. Moreover, they cannot generalize to real scans, which often contain topological noise (that changes the genus of the surface and makes the parameterization problem unsolvable).

Recent approaches tackle this problem by using Parameterization invariant data fidelity terms from Geometric measure theory (Bauer *et al.*, 2021a; Hartman *et al.*, 2022). A geodesic between two shapes A and B is a path that minimizes the Riemannian energy and for which both endpoints have 0 fidelity distance to A and B respectively. This formulation allows getting rid of the parameterization problem and to solve implicitly the reparameterization problem by directly matching endpoints to the surface. The promising results showed on different sets of unparameterized surfaces (high genus, topological changes, ...) open new possibilities for human shapes.

2.4 Geometric deep learning for human shapes

The field of geometric deep learning (Bronstein *et al.*, 2017) is a recent area. After the appealing results of convolutional neural networks (CNN) on images, it attempts to generalize the principles to geometrical data.

2.4.1 Deep learning for surfaces

Pioneer's works on deep learning for 3D Vision were a direct application of convolutions on 3D data. They can be applied on multi-view images (Su *et al.*, 2015), or as 3D CNNs (Gadelha *et al.*, 2017; Brock *et al.*, 2016) on volumetric data (density voxels replace pixels). However, while those approaches were successful on the task of surface classification, and more recently in partial shape reconstruction (Saint *et al.*, 2020; Chibane *et al.*, 2020), the need for computation resources for highly detailed shape makes them impracticable for deformation modeling.

PointNet (Qi *et al.*, 2016) has opened later a new way of applying deep neural networks by applying a Multi-Layer Perceptron (MLP) to each of the points before a max aggregation of the features, allowing to describe a point cloud easily. The architecture can be modified easily for point cloud segmentation and the MLP on 3D points is today an important piece of several works. PointNet++ (Qi *et al.*, 2017) applies the same idea to several local neighborhoods before global aggregation. The latter is nowadays still popular for point cloud segmentation or more. Several incrementations and improvements have been proposed in the community like KPConv (Thomas *et al.*, 2019), the Point Transformer (Zhao *et al.*, 2021) or the recent PointNext (Qian *et al.*, 2022).

Those approaches exploit the localization of surface points but do not take into account the mesh connectivity or the intrinsic geometry of the surface. Moreover, in order to be computationally efficient, they often need fixed-size nearest-neighbor sampling for defining local neighborhoods, thus being sensitive to parameterization.

The field of deep learning on surfaces, or meshes, tries to overcome this problem. A first idea was to use Graph Neural Networks (GNN), which has been proposed in several forms to tackle the problem (Boscaini *et al.*, 2015; Masci *et al.*, 2015; Monti *et al.*, 2017; Verma *et al.*, 2018): they allow to build operators that share similar properties than convolutions. Popular architecture like autoencoders or U-Net can then be generalized, while keeping the information of the connectivity structure (Bouritsas *et al.*, 2019; Ranjan *et al.*, 2018a; Zhou *et al.*, 2020b). However, original GNNs are *isotropic*, they aggregate surface information independently of the direction, as opposed to common CNNs. In order to improve the expressivity of those networks, several anisotropic filters have been proposed for meshes (De Haan *et al.*, 2020; Mitchel *et al.*, 2021). A few of them present the desired invariances in this thesis (Hanocka *et al.*, 2019; Mitchel *et al.*, 2021), but however are often computationally intensive and need to learn on reduced size meshes, thus lacking generalization with high sampling or noisy meshes. The Laplacian-based diffusion proposes to use the Laplacian to build a sampling invariant filter and showed interesting results to learn sampling invariants of shapes. However, the generalization of those recent filters to generative models is an open question. Indeed most neural architectures for human body generative models use older filters as their backbones, such as PointNet (Achlioptas *et al.*, 2018) or SpiralNet (Otberdout *et al.*, 2022).

2.4.2 Human body generative models

Generative models have known considerable progress in the deep learning era. GANs (Goodfellow *et al.*, 2014), VAEs (Kingma & Welling, 2014) and more recently diffusion models (Ho *et al.*, 2020), are powerful and generalizable generative models for 2D images. Their application to surfaces, and specifically to human bodies, opens the possibility to replace classical human shape spaces. Several autoencoders have been proposed in the context of mesh generation and were applied successfully to this problem. However the interpolation of two human surfaces is a hard problem and the architecture, or the training must ensure that the generated deformation is plausible. Multiple autoencoders have been proposed in the literature. The Neural3DMM (Bouritsas *et al.*, 2019) was one of the first to be able to overcome PCA in human mesh compression. Nowadays, the spectral autoencoder (Lemeunier *et al.*, 2022) architecture shows state-of-the-art results for mesh compression in an autoencoder latent space. However, no disentangling is done between human body shape and pose. Moreover, the spectral autoencoder, along with other counterparts, is not parameterization invariant.

In order to have parameterization invariance, the most popular approach nowadays is still to use the PointNet architecture. The GDVAE (Aumentado-Armstrong *et al.*, 2019) and LIMP (Cosmo *et al.*, 2020) approach superimposes different losses on deformation, separate in intrinsic and extrinsic quantities, in order to build a plausible latent space. Recent work demonstrates that incorporating ARAP-related losses in the training of latent deep learning models, for deformation (Huang *et al.*, 2021), human pose generation (Muralikrishnan *et al.*, 2022), or disentanglement (Zhou *et al.*, 2020a) encourages a better quality of output shapes. By combining, a very large dataset (more than 100000 shapes), a PointNet auto-encoder as an

initialization step, and a learned deformation model, the 3D-CODED approach and similar iterations were able to get state-of-the-art results on the FAUST human shape registration task. Finally, Hahner *et al.* (Hahner & Garcke, 2022) used hexagonal convolution for human pose auto encoding on semi-regular, which requires remeshing the data in a preprocessing step.

While those methods show great success in representing efficiently human shape spaces, their drawbacks come from their strong reliance on parameterized human body datasets on which they can overlearn information about surface parameterization. They moreover fail to learn efficiently the map from a linear space to a (non-flat) human shape space and thus do not generalize well when confronted with unseen data.

Representation	Tr	Sc	Rot	Param.	Training
Shape Distributions	I	N	I	≈	No
Extended Gaussian Images	I	N	N	✓	No
Shape Histograms	N	N	N	✓	No
Heat Kernel Signatures	I	N	I	✓	No
Wave Kernel Signature	I	I	I	✓	No
Area Projection Transform	I	I	I	✓	No
Extremal Human Curve	I	I	I	✓	No
ShapeDNA	I	N	I	✓	No
ARAP	I	N	I	✗	No
Elastic matching	I	N	I	✓	No
SMPL	I	I	I	≈	Yes
GDVAE	I	N	I	≈	Yes
Neural3DMM	N	N	N	✗	Yes
LIMP	N	N	N	≈	Yes
ARAPReg	I	N	N	≈	Yes

Table 2.1 A summary of a number of approaches for describing/deforming human shapes, showing their characteristic: whether they are (I)nvariant with respect to translation (Tr), scaling (Sc), and rotation (Rot), or whether they require (N)ormalization; their relative invariance to parameterization: it can be fully invariant, approximately invariant or discretization dependent; the need of training data.

Chapter 3

Mathematical definitions

Contents

As stated in the introduction, one of the challenges in the study of the human body comes from its representation and its parameterization. Its nature as a surface is easy to describe in words but is not an easy object to apprehend mathematically. We reintroduce in this section the most crucial mathematical notions and definitions that will be used in the manuscript. An interested reader is invited to look at the following books for more details (the list is, of course, non-exhaustive):

1. **Differential Geometry of Curves and Surfaces**, *Manfredo P. Do Carmo*, that introduces essential notions of geometry to understand surfaces
2. **An Introduction to Differentiable Manifolds and Riemannian Geometry**, *William M. Boothby*, that introduces more general notions about differentiable manifolds and the most essential tools (for the current manuscript) of Riemannian geometry
3. **Plateau's Problem – An invitation to Varifold geometry**, *Frederick J. Almgren, Jr.*, is a short book that introduces the varifold theory of surfaces

In this Chapter, we start by giving some mathematical definitions of a surface and some of its characteristics like the normal and curvatures. We then construct the shape space as the surface space quotiented by preserving transformations. After that, we introduce tools that we will use to work or leave the shape space: Varifolds, and Riemannian geometry. Finally, I want to emphasize that this is a computer science manuscript, where the notations and notions are introduced in order to serve later in the methodology and to help a new reader to understand properly the different approaches. They are thus simplified and often differ from the more complete and rigorous definition found in the maths literature.

3.1 Geometry of surfaces

3.1.1 Parameterized surfaces

When we speak about surfaces, we generally speak about a 2 dimensional manifold embedded in $\mathbb{R}^3$. Frankly speaking, this means that if we look *closely* enough to the surface, we are looking at a plane. The full mathematical definition is available in (do Carmo, 1976), but we will not use it in this manuscript. In fact, it would be hard to build a nice shape space from this purely local definition. Generally, authors prefer to work on parameterized surfaces, i.e, where we can define a global f that defines the surface.

Definition 3.1.1. *A **parameterized surface** is a smooth embedding from any open set $U \in R^2$ to $\mathbb{R}^3$, i.e. $f \in C^\infty(U, \mathbb{R}^3)$, $d_x f$ is injective for all $x \in U$, and f is injective.*

The image of f by U is called a **regular surface**, which we denote by S. f is called the **parameterization** of S.

The notion of parameterized surfaces can easily be used to define many concepts related to a surface S. The tangent space stems from the injectivity of the differential. Now for each $p \in S$, we will denote by $x_p \in U$ the x such that $f(x) = p$.

Definition 3.1.2. *The **tangent space** of S at p $T_p(S)$ is the image of $\mathbb{R}^2$ (the tangent space of U at x_p), by $d_{x_p} f$.*

Let $x_p = (u, v)$, the coordinates of x_p. We denote by f_u and f_v the partial derivatives of f in u and v. We can define the normal vector from the parameterization :

Definition 3.1.3. *Let $p \in S$. The **normal** of S at p is the unit vector defined by*

$$n(p) = \frac{f_u(x_p) \times f_v(x_p)}{|f_u(x_p) \times f_v(x_p)|}.$$

Now, if around each point, the normal vector is pointing in the same direction, we say that the surface is *oriented*. Some surfaces are not oriented, for example, the famous Möbius strip. Luckily for us, human shapes are and we can define normals globally.

Definition 3.1.4. *The application $N : S \to \mathbb{S}^2$, that maps any point p of S to its normal vector is called the **Gauss map**.*

This map has very nice properties – the Extended Gaussian Image presented in related work is the discretized version of the map – because it encodes information on the shape of the surface and can be used to extract efficient descriptors from the surface (chapter 4). Two very important properties are that it is *invariant* by the choice of parameterization and that it is *equivariant* by rotation, i.e, rotating the surface will rotate the Gauss map. However, the information encoded by the normal is mostly extrinsic, and the intrinsic information is encoded by the first fundamental form.

Definition 3.1.5. *The **first fundamental form** of S at p is a quadratic form on the tangent space $T_p(S)$. It is the restriction of the inner product on $\mathbb{R}^3$ onto $T_p(S)$, i.e. the **Riemannian metric** on S. We generally represent it as a positive definite symmetric 2×2 matrix g defined by:*

$$g(p) = \begin{pmatrix} E_p & F_p \\ F_p & G_p \end{pmatrix},$$

where

$$\begin{aligned} E_p &= \langle f_u(x_p), f_u(x_p) \rangle \\ F_p &= \langle f_u(x_p), f_v(x_p) \rangle \\ G_p &= \langle f_v(x_p), f_v(x_p) \rangle \end{aligned}$$

Its determinant $dA(p) = |f_u(x_p) \times f_v(x_p)|$ is called the area form.

The first fundamental form is essential because it allows for defining quantities on the surface such as lengths, areas, and angles between points of surfaces. The locally minimizing paths on the surfaces are called **surface geodesics** and generalize straight lines on surfaces. We will see later a nice visualization of the effect of metric changes on surfaces. The first fundamental thus contains the *intrinsic* information of the surface and is also invariant to rotations. Moreover, knowing the normal and the first fundamental form maps of a surface characterizes the surface up to translation :

Theorem 3.1.6. *(Fundamental theorem of surfaces)*
Two parameterized surfaces f_1 and f_2 having the same representation (g, n) differ at most by a translation (and rotation for g).

This theorem implies that we can represent a surface by its induced metric g and the unit normal field n, for the purpose of analyzing its shape. Comparing the first fundamental forms and normals will be crucial, especially when one wants to deform one shape into another.

However, note that this theorem cannot be applied straightforwardly to compare surfaces, since the expression of g is dependent on the chosen parameterization of the surface.

The last surface notion we need is a way of taking double derivatives.

Definition 3.1.7. *Let h a $\mathscr{C}^\infty$ function on S. Let $p \in S, v \in T_p(S)$ a unit vector on $T_p(S)$. For any α differentiable path on S such that $\alpha(0) = p$ and $\frac{d}{dt}\alpha(t) = v$, the vector $\frac{d}{dt}h(\alpha(t))$ is the same. We call this vector the **directional derivative** of a function h on S, in direction v, denoted d_v*

This definition allows only for taking the first derivatives. For second derivatives, we need to restrict the definition to *derivatives over a geodesic path*.

Definition 3.1.8. *Let h a $\mathscr{C}^\infty$ function on S. Let $p \in S, v \in T_p(S)$ a unit vector on $T_p(S)$. For any α geodesic path on S such that $\alpha(0) = p$ and $\frac{d}{dt}\alpha(t) = v$, we call the vector $\frac{d^2}{dt^2}h(\alpha(t))$ the **directional derivative** of order 2 of a function h on S, in direction v, denoted $d_v^2 h$*

This last definition allows us to define the Laplace-Beltrami operator generally used to derive functions on surfaces. It generalizes the Laplace operator to surfaces, which is in $\mathbb{R}^3$, $\Delta f = \frac{\partial^2 f}{dx^2} + \frac{\partial^2 f}{dy^2} + \frac{\partial^2 f}{dz^2}$ or the mean of all directional derivative of order 2. Multiple definition are possible for this operator, we rather chose the following definition, which is detailed in (Barthelmé, 2013), generalizing the idea of taking the mean of double derivatives :

Definition 3.1.9. *Let $p \in S$, h a $\mathscr{C}^\infty$ function on S. The **Laplace-Beltrami** operator of S at p is the mean over all directions, of directional derivatives of order 2 at point p.*

$$\Delta_p h = \frac{1}{2\pi}\int_{v\in\mathbb{S}^1} (d_v^2 f)(p) dv,$$

where we identify $\mathbb{S}^1$ to the space of all norm-1 tangent vectors in $T_p(S)$.

Knowing the notion of fundamental forms and normals, we would like now to compare surfaces between them by embedding them in a surface space. To define a surface space, it is generally preferred to work with a common model from which to define surfaces. In this thesis, we choose to see each surface as a map from a human template $\mathscr{T}$. The chosen template is displayed in Figure 3.1. We can generalize the notions of differentiable maps from a surface and thus define human surfaces as differentiable maps from a single template. We thus work with what we call template-parameterized surfaces.

Definition 3.1.10. *A **template-parameterized surface** is a smooth map from a template surface $\mathscr{T}$ to $\mathbb{R}^3$ that is an embedding, i.e.*

$f \in C^\infty(\mathscr{T}, \mathbb{R}^3)n$, and f is injective. We denote by $\bar{f}$, the template surface, i.e. the identity.

All notions presented above can be generalized to template parameterized surfaces. It is done by associating some local coordinates (u, v) around on each point of the template. A nice

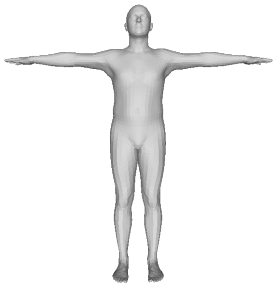


Figure 3.1 The chosen template

thing when we have a common model from which surfaces are defined is that one can better see the effects of changing the intrinsic geometry of a surface, i.e. its fundamental form. In Figure 3.2, we visualize two surfaces of a different identity, parameterized by the template on the left. On each shape, we compute the geodesic paths between the right arm and the left of the belly. We then look at the corresponding paths on the template. Changing identity equals changing the intrinsic geometry of the surface, thus the paths are different.

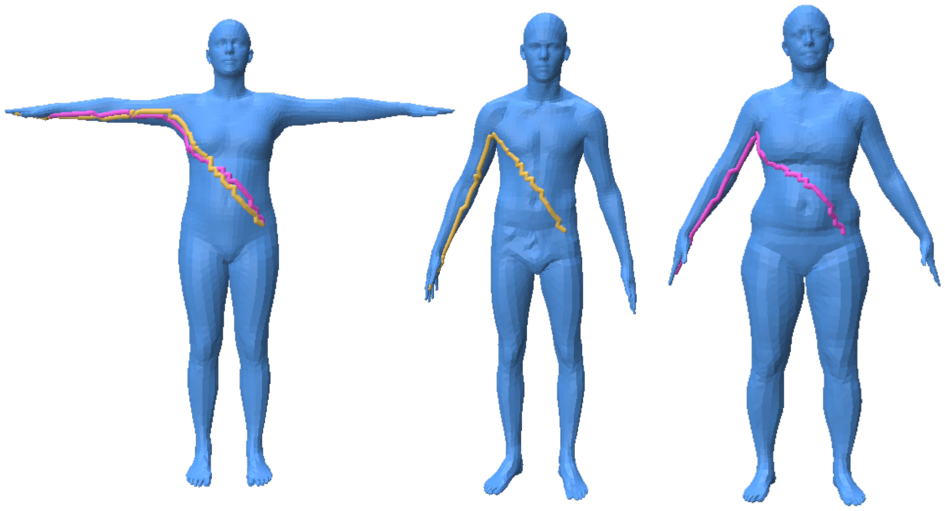


Figure 3.2 Two shapes (right) in correspondence with a template (left). We compute on those shapes the geodesic path between the same points on the template. Mapping back those paths to the template, we see that they are different. The metric has changed between the two shapes.

Finally, having a common model from which surfaces are defined allows us to define a surface space.

Definition 3.1.11. *The* ***surface space*** $\mathcal{S}$ *is the space of all embedings from a template* $\mathcal{T}$

While this space has the nice property to be a differentiable manifold, it does not give correspondences between two surfaces. This space is often called *pre-shape space*, because two parameterized surfaces can still represent the same regular surface S, for example, if there is a reparameterization between them, Figure 3.3. Thus, we do not want to work directly with the surface space, but will introduce the transformations that preserve a surface, before giving our final definition of human body shape space.

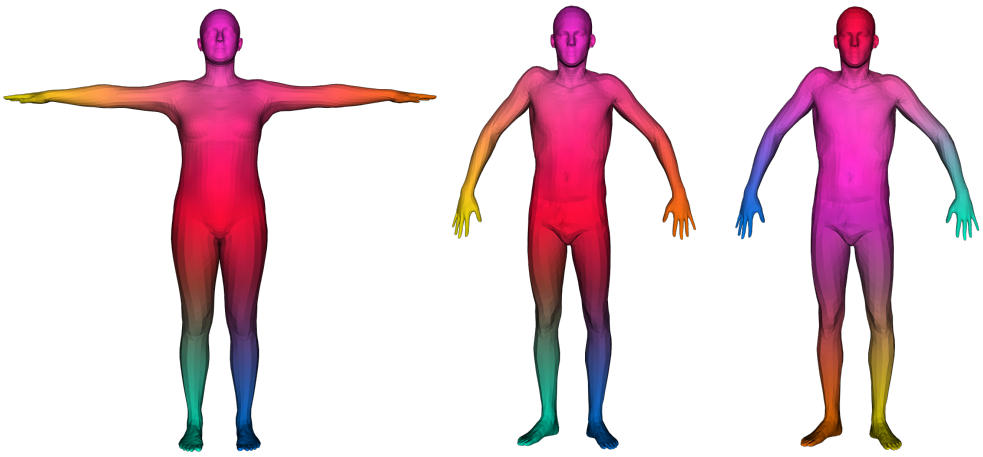


Figure 3.3 A template surface, and 2 parameterized surfaces representing the same object. In this case, the 2 parameterized surfaces differ by a reparameterization.

3.1.2 Actions on surfaces

As stated in the introduction, some transformations are applied during the collection of human body data. Those transformations do not change the shape itself, but they will change their numerical representation. We say that they are *preserving* transformations, and we view them mathematically as actions on the surface space. We define below the main transformations encountered.

Definition 3.1.12. *Let G be a group and X a set. G is said to act on X (in this thesis, mostly, on the left) if there is a mapping $\theta : G \times X \to X$, such that:*

1. *If e is the identity element of G, then, $\theta(e,x) = x$*
2. *If $g_1, g_2 \in G$, then $\theta(g_1, \theta(g_2,x)) = \theta(g_1 g_2, x)$*

The mapping is called a ***group action****. We also denote the action of G on X by $(g,x) \to g.x$.*

Several group actions act on the surface :

Definition 3.1.13. *The group $\mathbb{R}^+$, acts by* **scaling** *on the surface space via the mapping $(\beta, f) \to \beta f$*

Definition 3.1.14. *The group $\mathbb{R}^3$, acts by **translation** on the surface space via the mapping $(T, f) \rightarrow f + T$*

Definition 3.1.15. *The group $SO(3)$, acts by **rotation** on the surface space via the mapping $(R, f) \rightarrow Rf$*

Definition 3.1.16. *The group $Diff^+(\mathcal{T})$ of oriented preserving diffeomorphisms acts by **reparameterization** on the surface space via the mapping $(\gamma, f) \rightarrow f \circ \gamma^{-1}$*

Those 4 actions can be treated separately, or together, by composing the actions, obtaining a product group G. Knowing the action of some group G, we can now define equivalence relations between two surfaces f_1, f_2, as $S_1 \sim S_2$ if there exists a $g \in G$ such that $f_1 = g.f_1$.

Definition 3.1.17. *For a surface f, the set $[f]$ of surface in equivalence with f is called the **equivalence class** of f*

In particular, when G is the reparameterization group $\text{Diff}^+(\mathcal{T})$, the equivalence of $f \in \mathcal{H}$ is characterized by all surfaces such that $f(\mathcal{T}) = S$, i.e. the elements in $[f] = \{f \circ \gamma^{-1} \text{ for } \gamma \in \Gamma\}$ are all possible registrations of S. We can then define the *quotient space* of the surface space by G, which is the space of all equivalence classes of surfaces.

Definition 3.1.18. *Let G be the chosen group of preserving transformations of surfaces. The **human body shape space** $\mathcal{H}$ is the space of all human body surfaces quotiented by the action of G. In other terms, it is the space of equivalence classes.*

G is taken to be the product group $\mathbb{R}^+ \times \mathbb{R}^3 \times SO(3) \times \text{Diff}^+(\mathcal{T})$, however in some cases, some actions are quotiented during data acquisition, and we work with only a subset of G.

We have now defined our shape space, which we will use in the manuscript. Depending on the desired objective, we will need to leave the shape space to build descriptors of human bodies (chapter 3 and 4), or rather consider it as a Riemannian manifold, by equipping it with a Riemannian metric in order to see deformation as a path on this manifold (chapter 5 and 6).

3.1.3 Discretized surfaces

The previous definitions are useful to define surfaces as objects and define quantities that will serve to describe or deform them. However, they are not the exact same objects in a computer. We present here the discrete counterparts that will serve for numerical implementations of algorithms.

Definition 3.1.19. *A **surface mesh** M is a piecewise oriented linear surface, represented by a simplicial complex.*

In other terms, a surface mesh is a polyhedral surface for which every face is a triangle. Moreover, any two adjacent faces share the same orientation.

Formula of normal and area of faces
On each of these faces, we can easily compute the normal of a triangle and its area :

Let p_1, p_2, p_3, the positively ordered points of a face, $e_{21} = p_2 - p_1, e_{31} = p_3 - p_1, e_{32} = p_3 - p_2$. The normal of the face is defined as :

$$n = \frac{e_{21} \times e_{31}}{|e_{21} \times e_{31}|}$$

The area of the face is $A = \frac{1}{2}|e_{21} \times e_{31}|$

Thus, in a mesh, each triangle face's tangent plane is the triangle plane. Generally, the normal (or tangent space.) at a vertex (or at an edge) is simply the mean of normal vectors over adjacent faces. We now wish to compute the first fundamental form of the mesh.

Computation of the first fundamental form

In this manuscript, we use the first fundamental form to compare surfaces in Chapters 6 and 7. The values depend on which parameterization we chose. In chapter 6, we will have a preferred parameterization for each surface, and define the surface metric on the template directly. On the opposite, in chapter 7, we work without a preferred parameterization (no surface correspondence). We will define the surface metric as the metric restricted to the triangles.

First fundamental form of a surface on the template

We compute the first fundamental form of each triangle. Let t_i be a triangle of the template mesh. Let its vertices be p_1, p_2, p_3 on the template itself. We take local coordinate frames (u, v) in the plane, such that p_1 equals to $(0, 0)$, p_2 to $(u_2, 0)$, p_3 to (u_3, v_3).

Now given a mesh M in correspondence with the template, let t_i^M be the triangle that corresponds to t_i, with vertices p_1, p_2, p_3 on M. In order to compute the first fundamental form, we need to find the affine f that maps the triangle p_1, p_2, p_3 to q_1, q_2, q_3 (see Figure 3.4 for a visual explanation).

Proposition 3.1.20. *The derivatives of f are given by:*

$$\frac{\partial f(u,v)}{\partial u} = \frac{1}{l_1}(q_2 - q_1)$$

$$\frac{\partial f(u,v)}{\partial v} = \frac{\tan\theta}{l_1}(q_1 - q_2) + \frac{1}{H}(q_3 - q_1),$$

where H is the height of the triangle, when the edge $p2 - p1$ is taken as the basis of t_i, and θ is the angle between $p2 - p1$ and $p3 - p1$. We compute the first fundamental form by taking the inner products of those quantities.

Proof. Given u, v, f has the form:

$$f(u,v) = \lambda_1(u,v)q_1 + \lambda_2(u,v)q_2 + \lambda_3(u,v)q_3$$

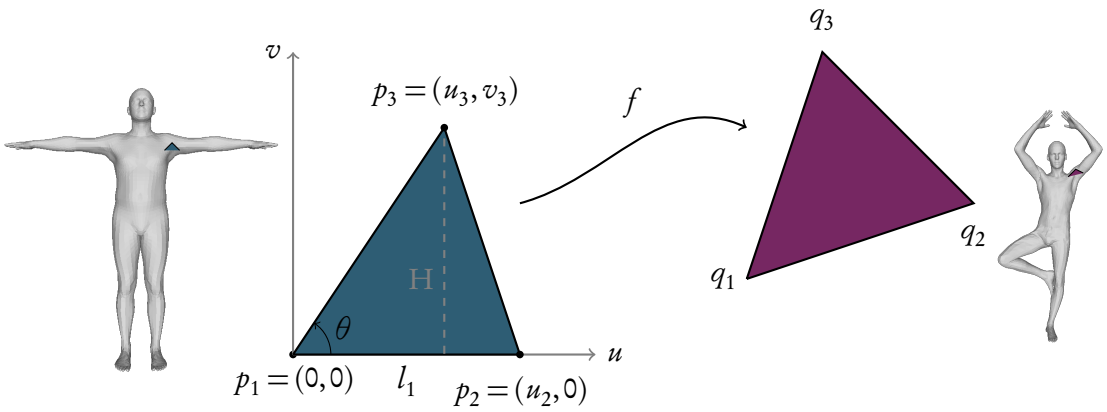


Figure 3.4 Ordered triangle with their corresponding local reference frame for the numerical computation of first fundamental form. Given a human shape (on the right), we search for the first fundamental form of the pink triangle. The corresponding triangle on the template (left) is the blue triangle for which we take local coordinates as in the figure. The quantities l_1, H, θ are useful in the computation of the derivatives.

Where $\lambda_1(u,v), \lambda_2(u,v), \lambda_3(u,v)$ are the barycentric coordinates of a point (u,v) in t_i. They respect the conditions $\lambda_1 + \lambda_2 + \lambda_3 \leq 1, \lambda_i = 0$, and in t_i, they are such that u and v are equal to :

$$\begin{cases} u = \lambda_1(p_1)_u + \lambda_2(p_2)_u + \lambda_3(p_3)_u = & \lambda_2 u_2 + \lambda_3 u_3 \\ v = \lambda_1(p_1)_v + \lambda_2(p_2)_v + \lambda_3(p_3)_v = & \lambda_3 v_3 \end{cases}$$

In other terms, the λ_i are defined as the solution of the following equation (see (Lévy, 2008) for similar calculations):

$$\begin{pmatrix} 0 & u_2 & u_3 \\ 0 & 0 & v_3 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix} = \begin{pmatrix} u \\ v \\ 1 \end{pmatrix}$$

The solution of this equation is given by:

$$\begin{aligned} \lambda_1 &= \frac{u_2 v_3 - v_3 u + v(u_3 - u_2)}{u_2 v_3} \\ \lambda_2 &= \frac{v_3 u - u_3 v}{u_2 v_3} \\ \lambda_3 &= \frac{v}{v_3} \end{aligned}$$

The derivatives of f are given by:

$$\begin{aligned} \frac{\partial f(u,v)}{\partial u} &= \frac{\partial \lambda_1}{\partial u} q_1 + \frac{\partial \lambda_2}{\partial u} q_2 + \frac{\partial \lambda_3}{\partial u} q_3 \\ \frac{\partial f(u,v)}{\partial v} &= \frac{\partial \lambda_1}{\partial v} q_1 + \frac{\partial \lambda_2}{\partial v} q_2 + \frac{\partial \lambda_3}{\partial v} q_3, \end{aligned}$$

which becomes :

$$\frac{\partial f(u,v)}{\partial u} = \frac{1}{u_2}(q_2 - q_1)$$
$$\frac{\partial f(u,v)}{\partial v} = \frac{u_3}{v_3 u_2}(q_1 - q_2) + \frac{1}{v_3}(q_3 - q_1)$$

The value of u_2 is simply the length of the first edge of the triangle (l_1). u_3 and v_3 are the projection of first and second edge on the u-axis and v-axis. So v_3 is the height H of the triangle (relative to the first edge as the basis), and $\frac{u_3}{v_3}$ is $\tan(\theta)$, where θ is the angle between the first edge and second edge (see Figure 3.4 for a better understanding of those quantities). □

The value of g comes naturally by taking the inner products of partial derivatives, using the formula in the definition of the first fundamental form.

First fundamental form from the ambient space
This case gives much simpler computations. In this case, we take for each triangle t_i of the template mesh, the local coordinates (u,v), such that p_1 equals to $(0,0)$, p_2 to $(1,0)$, p_3 to $(0,1)$ (see Figure 3.5 for a visual explanation).

Proposition 3.1.21. *The partial derivatives are given by:*

$$\frac{\partial f(u,v)}{\partial u} = (p_2 - q_1)$$
$$\frac{\partial f(u,v)}{\partial v} = (p_3 - q_1)$$

By taking the inner products, we obtain a very simple formula given by (see (Cheeger et al., *1984) in Appendix 3 for more general computations) :*

$$g_i = \begin{pmatrix} l_{12}^2 & \frac{1}{2}(l_{12}^2 + l_{13}^2 - l_{23}^2) \\ \frac{1}{2}(l_{12}^2 + l_{13}^2 - l_{23}^2) & l_{13}^2 \end{pmatrix},$$

where l_{ij} is the length of $q_i - q_j$.

Proof. The solution is slightly easier, as the affine f that maps t_i to q_1, q_2, q_3 is written as $f(u,v) = uq_2 + vq_3 + (1-u-v)q_1$. The formula of derivatives is straightforward. □

Laplace Beltrami cotangent formula
For an N-vertex mesh M, the cotangent Laplace operator is represented by a $N \times N$ symmetric matrix L. Let $e = (p_i, p_j)$ be an edge of the mesh. We denote by α_{ij} and β_{ij} the angles of the opposite sides of the two adjacent triangles of e. The cotangent formula of the Laplace operator applied to a scalar function u on M at vertex i is given by:

$$(\Delta u)_i = \frac{1}{2A_i} \sum_j (\cot(\alpha_{ij}) + \cot(\beta_{ij}))(u_i - u_j)$$

Where the sum is done over all adjacent edges to vertex i.

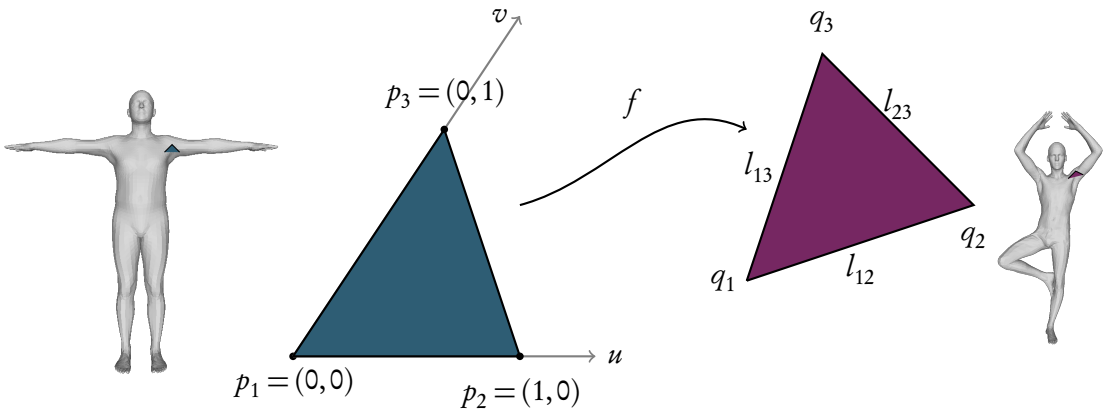


Figure 3.5 Ordered triangle with their corresponding local reference frame for the numerical computation of first fundamental form. Given a human shape (on the right), we search for the first fundamental form of the pink triangle. The corresponding triangle on the template (left) is the blue triangle for which we take local coordinates as in the figure. The quantities l_{12}, l_{13}, l_{23} are part of the final formulation.

3.2 Surfaces as geometric measures

We present here the main elements that will later allow us to leave the shape space to practical spaces for human shape comparison.

The geometric measure theory was raised to solve Plateau's problem, named after Joseph Plateau's experiments on soap films (Plateau, 1873). This problem is out of the scope of this thesis, we present in this section a simplified notion of varifold that is broad enough for this manuscript. We will also rapidly introduce their discretizations in the case of triangular meshes.

Definition 3.2.1. *Let L be a linear space. The **dual space** L^* is the space of **linear functionals**, ie functions such that $f : L \to \mathbb{R}$ such that f is linear.*

Definition 3.2.2. *Let L be a linear space endowed with a norm $||.||$. A linear functional μ over L is a **continuous** element of L^*, i.e. a bounded element of L^*: $\exists K, \forall x \in L, \mu(x) \leq K||x||$.*

Definition 3.2.3. *Let X be a topological space. A **measure on the space of functions** $(\mathscr{C}_0(X, \mathbb{R}), ||.||_\infty)$ is a linear functional over $\mathscr{C}_0(X, \mathbb{R})$.*

Varifolds are specific measures, where the set X is taken as $\mathbb{R}^3 \times \mathbb{S}^2$.

Definition 3.2.4. *A (oriented) **varifold** μ is a measure over the space of functions on the product space $\mathbb{R}^3 \times \mathbb{S}^2$ (coordinates and orientation), $(\mathscr{C}_0(\mathbb{R}^3 \times \mathbb{S}^2, \mathbb{R}), ||.||_\infty)$.*

Every surface S defines a varifold μ_S, evaluated on $\phi \in \mathscr{C}_0(\mathbb{R}^3 \times \mathbb{S}^2, \mathbb{R})$ as follows:

$$\mu_S(\phi) = \int_S \phi(x, n(x)) dA(x)$$

An important feature of this formulation is that it is *independent* of the parameterization of the surface. It is the main reason for their use in shape analysis. In the discretized case of a surface mesh M, each face t_i is seen as an area-weighted Dirac varifold $a_i \delta_{(c_i, n_i)}$. The varifold

μ_M associated to M is simply the sum of these measures: $M = \sum_{i=1}^{m} a_i \delta_{(c_i,n_i)}$. The application of μ_M to a function ϕ is done by:

$$\mu_M(f) = \sum_{i=0}^{n_f} a_i \phi(c_i, n_i)$$

Note that this discretization is only robust to the remeshing (reparameterization) of a surface. For example, the subdivision of a triangle T into two triangles T_1, T_2 is such that $\mu_T \neq \mu_{T_1} + \mu_{T_2}$ (see Figure 3.6 for an illustration of the phenomenon).

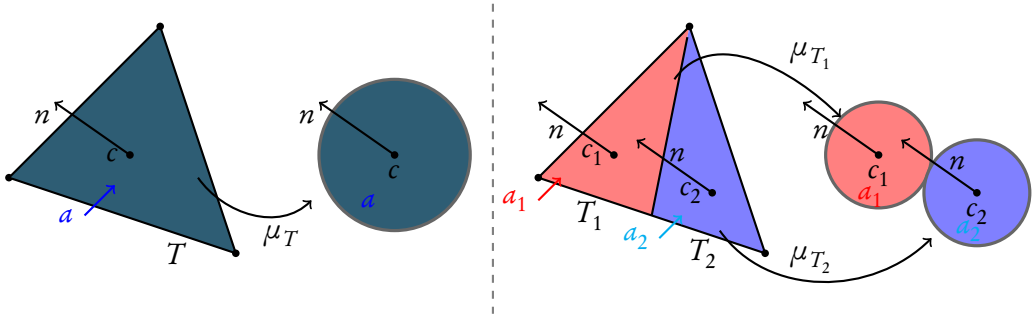


Figure 3.6 Varifold of triangles in the case of discretized meshes. On the left, triangle T is discretized into $\mu_T = a\delta_{(c,n)}$, that we represent in the following way: we keep the center and the normal; the area weight a is represented by a surrounding disk of area proportional to a. On the right, the triangle T_1, T_2 are discretized into $\mu_{T_1} = a_1\delta_{(c_1,n_1)}$ and $\mu_{T_2} = a_2\delta_{(c_2,n_2)}$. The sum of those two measures is different from μ_T.

However, in the case where triangles are sufficiently small, this change will be undistinguishable and by some abuse of language, we say that the discretization of varifold is invariant to remeshing of surfaces.

Another advantage of seeing surfaces as varifolds are that it allows embedding surfaces in a linear space. The common approach to compare varifolds in the shape analysis literature (Charon & Trouvé, 2013; Kaltenmark *et al.*, 2017) is to use the theory of Reproducing Kernel Hilbert Spaces.

3.3 Reproducing Kernel Hilbert Space

The theory of Reproducing Kernel Hilbert Spaces (RKHS) is a very popular tool of modern machine learning. It is useful because it allows linearizing some hard classification problems. We define the main elements and theorems that allow the use of this theory in the context of the varifold theory for surfaces.

Definition 3.3.1. *The pair* $(H, \langle\ ,\ \rangle)$ *is a* ***Hilbert space*** *if*

- H *is a linear space on* $\mathbb{R}$
- *The mapping* $\begin{cases} \langle\ ,\ \rangle : H \times H \to \mathbb{R} \\ x, y \mapsto \langle x, y \rangle \end{cases}$ *is an inner product*

- *H is complete relative to the topology that arises from the distance function defined by the inner product $\langle\ ,\ \rangle$*

Definition 3.3.2. *A **semi-positive definite (SPD) kernel** K on a set X is a mapping $X \times X \to \mathbb{R}$ such that:*

- $K(x,y) = K(y,x),\ \forall x,y \in X,$
- *If $x_1,...,x_n \in X,\ \lambda_1,...,\lambda_n \in \mathbb{R}$, then $\sum_{i=1}^{n} K(x_i,x_j)\lambda_i\lambda_j \geq 0$.*

Several SPD kernels exist and are popular in the literature, among which :

- The inner product between two elements of $\mathbb{R}^n$.
- Let $x,y \in \mathbb{R}^n, \sigma \in \mathbb{R}$, the kernel defined by $k(x,y) = \exp(-\frac{||x-y||^2}{\sigma^2})$.

It is also possible to combine kernels :

- Let K_1, K_2 two SPD kernels, $\lambda,\beta \geq 0$. Then $\lambda K_1 + \beta K_2$ is a SPD kernel
- Let K_1, K_2 two SPD kernels. The product K_1K_2 is a SPD kernel

Theorem 3.3.3. *(Riesz representation theorem) If $\xi : H \to \mathbb{R}$ is a bounded linear functional, then there exist $w \in H$ such that*

$$\xi(x) = \langle x,w\rangle, \forall x \in H$$

Definition 3.3.4. *The pair $(H,\langle\ ,\ \rangle)$ is a **Reproducing Kernel Hilbert space (RKHS)** if*

- *$(H,\langle\ ,\ \rangle)$ is space of functions, i.e. there exists a space X such that any $f \in H$ is such that f is a function $X \to \mathbb{R}$*
- *For all $x \in X$, the linear functional (element of the dual space) on H $\xi_x f \mapsto f(x)$ is bounded*

Let $x \in X$, based on the Riesz representation theorem, there exists $K_x \in H$ such that ξ_x defined as above can be rewritten to $\xi_x(f) = \langle f, K_x\rangle$. The function:

$$\begin{cases} K : X \times X \to \mathbb{R} \\ \qquad x,y \mapsto K_x(y) = K(x,y) \end{cases}$$

is a positive definite kernel. K is called a Reproducing Kernel.

Theorem 3.3.5. *(Aronzajn, Moore) Let $K : X \times X \to \mathbb{R}$, an SPD kernel, then there exists a Hilbert space H of functions on X such that K is a Reproducing Kernel on H.*

The idea behind using varifolds for surface shapes originally presented in (Charon & Trouvé, 2013) and (Kaltenmark *et al.*, 2017) is to map the space of surfaces, seen as varifolds, to a Reproducing Kernel Hilbert space of functions $\mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$, thanks to an SPD kernel on $\mathbb{R}^3 \times \mathbb{S}^2$. We will use this idea in chapter 5 to construct descriptors of human motion. In the contrary, in chapter 4, we map the space of human body surfaces to the space of functions $\mathbb{S}^2 \to \mathbb{R}$, which allows building descriptors of human pose.

3.4 Riemannian geometry of the shape space

We define some elements of Riemannian geometry we will need in the last chapters. We view the parameterized surface space and the shape space as differentiable manifolds. The idea is similar to surfaces: if we look closely enough at a differentiable manifold, we will see a flat space, i.e a linear space L. We recommend the definition given in (Boothby, 2003, Chapter 5).

Some basic differentiable manifolds are:

- The Euclidean space $\mathbb{R}^n$,
- The sphere $\mathbb{S}^2$,
- The rotation group $SO(3)$, which is a differentiable manifold of dimension 3,
- The parameterized surface space.

Tangent spaces are generalizations of the notion of the tangent plane for the surface. For each $\mathscr{C}_1$ path on a manifold, we can compute its derivative, a tangent vector to the manifold. The tangent space at point p, $T_p(M)$ is the set of all possible tangent vectors at point p. $T_p(M)$ is a vector space.

With the notion of tangent space, we can define it similarly as for surfaces, a local inner product, called the Riemannian metric, that allows for generalizations of lengths, volumes, and angles.

Definition 3.4.1. *A **Riemannian metric** on a manifold M is a smooth Euclidean inner product g_p on $T_p(M)$. This means first that for any each $p \in M$, g_p is a positive definite bilinear mapping $T_p(M) \times T_p(M) \to \mathbb{R}$ (inner product). Also, for any X and Y two smooth vector fields, the inner product $g_p(X_p, Y_p)$, often denoted as $\langle X_p, Y_p \rangle_p$, is a smooth function of p.*

*A differentiable manifold with a Riemannian metric is called a **Riemannian manifold**.*

The differentiable manifolds presented before are all Riemannian manifolds with the following inner products :

- $\mathbb{R}^n$ with the Euclidean inner product,
- The sphere, with the restriction of $\mathbb{R}^3$ inner product to the tangent space,
- The tangent space of the rotation group $T_R SO(3)$ at point R is given by $\{R\Omega | \Omega \in \mathbb{R}^3, \Omega = -\Omega\}$, ie the set of skew-symmetric matrices multiplied by R. Those matrices live in $\mathbb{R}^{3\times 3}$ and can be equipped with the restriction of the matrix-inner product,
- The surface space tangent space is isomorphic to $\mathscr{C}_\infty(\mathscr{T}, \mathbb{R})$. We can endow it with the following L_2 product:

$$((h,k))_f = \int_{\mathscr{T}} \langle h(p), k(p) \rangle dA(p),$$

where dA is the area form of f and $h, k \in T_f(\mathscr{S}) = \mathscr{C}_\infty(\mathscr{T}, \mathbb{R})$. We can also use more complex Sobolev metrics, such as :

$$((h,k))_f = \int_{\mathscr{T}} \langle h(p), k(p)\rangle + \mathrm{tr}(dh(p)(g_f(p)^T g_f(p))^{-1} dk(p)^T) + \langle \Delta_f h(p), \Delta_f k(p)\rangle dA(p) \tag{3.1}$$

for h a vector of the tangent space T_f, i.e. a vector field on the template $\mathscr{T}$. dh and dh are the per-coordinate derivatives of h and k on f, expressed in a local coordinate system (u, v) :

$$dh = \begin{pmatrix} d_u h_x & d_v h_z \\ d_u h_y & d_v h_y \\ d_u h_z & d_v h_z \end{pmatrix}, \quad dk = \begin{pmatrix} d_u k_x & d_v k_z \\ d_u k_y & d_v k_y \\ d_u k_z & d_v k_z \end{pmatrix}$$

The second term $\mathrm{tr}(dh(p)(g_f(p)^T g_f(p))^{-1} dk(p)^T)$ is the trace of a 3×3 matrix induced by the product of 4 matrices of size $3 \times 2,\ 2 \times 2,\ 2 \times 2, 2 \times 3$. It can be viewed as the inner product of $dh(p)$ and $dk(p)$, relative to the metric of the surface f. A decomposition into interpretable terms is given in Chapter 7.

We can also use immersions between manifolds to define new Riemannian metrics

Definition 3.4.2. *(Pullback metric) Let $F : M \to N$ an immersion between two Riemannian manifolds M and N and $\langle , \rangle$ a Riemannian metric on N, then the bilinear form on $T_p(M)$ defined by*

$$\left(X_p, Y_p\right)_p = \langle dF(X_p), dF(Y_p)\rangle_{F(p)},$$

where $X_p, Y_p \in T_p(M)$ and $dF : T_p(M) \to T_p(N)$ is the differential of F. This metric is called the pullback metric of the Riemannian metric $\langle\ ,\ \rangle$ by F

Definition 3.4.3. *Let $\alpha : [0,1] \to M$ a ($\mathscr{C}^1$) **path** on the Riemannian manifold M. We define the **energy** of the path $E(\alpha)$ and the **length** of the path $L(\alpha)$ as :*

$$E(\alpha) = \int_0^1 \left\langle \frac{d\alpha(t)}{dt}, \frac{d\alpha(t)}{dt} \right\rangle dt, \quad L(\alpha) = \int_0^1 \sqrt{\left\langle \frac{d\alpha(t)}{dt}, \frac{d\alpha(t)}{dt} \right\rangle} dt,$$

Definition 3.4.4. *Let $p_0, p_1 \in M$ two points in a Riemannian manifold M. We define the **Riemannian distance** between p_0 and p_1, $d(p_0, p_1)$, as the length of the infimum of the length of all possible paths between p_0 and p_1.*

$$d(p_0, p_1) = \inf_{\alpha \in \mathscr{P}_{p_0}^{p_1}} L(\alpha), \tag{3.2}$$

where $\mathscr{P}_{p_0}^{p_1}$ is the set of all paths starting at p_0 and ending at p_1 : $\mathscr{P}_{p_0}^{p_1} = \{\alpha | \alpha : [0,1] \to M, \alpha(0) = p_0, \alpha(1) = p_1\}$

Very few Riemannian manifolds have closed-form solutions for a distance formula :

- The space of coordinates, with the natural distance, spanned by the inner product,

- The sphere, with the distance: $d(n_1, n_2) = \arccos(n_1, n_2)$,
- The rotation group with the distance: $d(R_1, R_2) = \arccos\left(\frac{tr(R_1 R_2^T - 1)}{2}\right)$.

We can define the curves that minimize locally the distances.

Definition 3.4.5. *A **geodesic segment** on a Riemannian manifold M is a curve $\alpha : [a, b] \to M$ for that minimizes the Riemannian distance for any sub-interval of $[a, b]$. This means that for any $t_1, t_2 \in [a, b]$:*

$$L(\alpha)_{[t_1,t_2]} = d(\alpha(t_1), \alpha(t_2)) = |t_1 - t_2|$$

Definition 3.4.6. *A curve $\alpha : I \to M$ is a **geodesic** if :*

$$\forall t \in I,\ \exists \epsilon > 0\ s.t\ \alpha : [t - \epsilon, t + \epsilon] \to M \text{ is a geodesic segment.}$$

A geodesic is a minimizer of the Riemannian distance and the Riemannian energy.

There is no closed-form solution for distances on the surface space. Computing the distance between two surfaces will require searching for the geodesic path at the same time. In practice, it is numerically more practical to first minimize the energy to obtain the geodesic before computing distances. This is not necessarily a problem when searching for deformations, but becomes highly unpractical in the case of pure surface comparison.

We will now introduce a metric on the shape space. The shape space is not really a differentiable manifold (taking quotient spaces can induce some mild singularities), but most notions of differentiable manifolds apply to it. In particular, it has a tangent space that we can equip with a Riemannian metric and thus introduce distances.

The common way to do this is to use submersive metrics.

Definition 3.4.7. *Let M and N be two differentiable manifolds. The application $\Pi : M \to N = M/G$ is a submersion if p is surjective, differentiable, and $D_x\Pi$ is surjective for any $x \in M$*

Let M be a differentiable manifold and G be a group that acts on M, in such a way that the application $\Pi : M \to N = M/G$ defined by $\Pi(x) = [x]$ is a submersion. For $x \in M$, the tangent space $T_x M$ is decomposed in a sum of two vector spaces: $\mathrm{Ker} D_x\Pi$, and its orthogonal, the *horizontal* space $(\mathrm{Ker} D_x\Pi)^\perp$. The map $D_x\Pi$ is a linear isomorphism (bijective) between $(\mathrm{Ker} D_x\Pi)^\perp$ and $T_{[x]}N$. In other words, for each $v \in T_{\Pi(x)}N$, there exists a unique $\tilde{v} \in (\mathrm{Ker} D_x\Pi)^\perp$, such that $D_x\Pi(\tilde{v}) = v$ (also named the horizontal lift of v). Figure 3.7 depicts visually the idea on a simple cylinder example.

Now if we have a Riemannian metric on M, we almost have a way to define $\langle w, v \rangle_{\Pi(x)}$ by $\langle \tilde{w}, \tilde{v} \rangle_x$. The problem is that the choice may depend on x. Equivalently, the metric on M must be invariant under the action of G.

In our case, the tangent space of the surface space is also a space of functions, $\mathscr{C}^\infty(\mathscr{T}, \mathbb{R}^3)$. The group actions on the surface space are the same in the tangent space. In this case, the condition for invariance takes the form :

$$((g.h, g.k))_{g.f} = ((h, k))_f, \forall h, k \in T_f, g \in G, f \text{ surface}$$

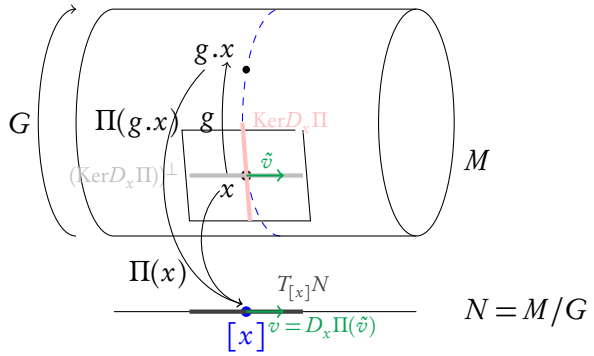


Figure 3.7 Illustration of the idea of submersive metrics. We display a cylinder M, seen as a Riemannian manifold, and a group action G, which is the group of rotations around the cylinder's axis. The quotient $N = M/G$ is a segment. Now, since the projection Π from M to N is a submersion, for $v \in T_{\Pi x}N$, there exists a unique $\tilde{v} \in (\text{Ker}D_x\Pi)^\perp \subset T_xM$, such that $(D_x\Pi)(\tilde{v}) = v$. We w wish to use the Riemannian metric on M to define one on N, by setting the inner product $\langle v, w\rangle_{[x]}$ between $v, w \in T_{[x]}N$ to be the inner product $\langle \tilde{v}, \tilde{w}\rangle_x$ on T_xM.

The research field that uses this approach to define Riemannian metrics on the shape space is called Elastic Shape Analysis (ESA).

We now have distances and geodesics on our shape space. However, working directly on the shape space is unpractical. In practice, to look for geodesics on the shape space, we look for *horizontal* geodesics on the surface space. The velocity of these geodesics $\alpha*$ are horizontal and their length on the surface space equals the length of their projections $[\alpha*]$ on the shape space.

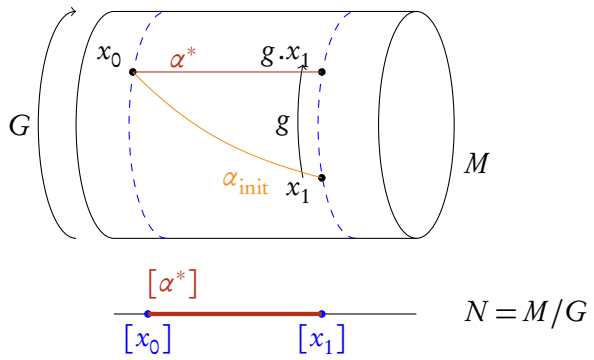


Figure 3.8 Illustration: searching for geodesics when using submersive metrics. We use the cylinder M of last example. We wish to find the geodesic $[\alpha^*]$ between $[x_0]$ and $[x_1]$. The geodesic α_{init} between the chosen representatives of x_0 and x_1 does not project horizontally to N. We thus search for a g such that the geodesic between x_0 and x_1 realizes this condition, obtaining the path α^*, whose length is $d([x_0],[x_1])$.

In other terms, the distance between two shapes can be defined as the infimum of the length :

$$d([f_0],[f_1]) = \inf_{g\in G} \inf_{\alpha\in\mathscr{P}_{f_0}^{g\cdot f_1}} \int_0^1 \sqrt{((\partial_t\alpha(t),\partial_t\alpha(t)))_{\alpha(t)}}\,dt,$$

where the infimum is taken over both paths α and transformation g. We illustrate the idea in Figure 3.8.

In general, we rather search for the geodesics α by minimizing the energy:

$$\inf_{g \in G} \inf_{\alpha \in \mathscr{P}_{f_0}^{g \cdot f_1}} E(\alpha) = \inf_{g \in G} \inf_{\alpha \in \mathscr{P}_{f_0}^{g \cdot f_1}} \int_0^1 ((\partial_t \alpha(t), \partial_t \alpha(t)))_{\alpha(t)} dt \tag{3.3}$$

The distance is then computed directly using the length formula. Finding the solution to this problem is the core difficulty of ESA, and Chapters 5 and 6 will present some ways to overcome it.

Chapter 4

Projection-based Classification of Surfaces for 3D Human Mesh Sequence Retrieval

4.1 Introduction

The problem of 3D shape and 3D shape sequence retrieval is a hard problem because of the difficulty of surface comparison. Having different discretizations between shapes causes this difficulty. It is further complicated by the need, in retrieval, to identify surfaces that differ only by Euclidean transformations and similarities.

The way we define our shape space in the previous Chapter allows in theory, the perfect or nearly perfect comparison of large classes of shapes. Nevertheless, there are many situations where we would need or prefer a quicker and rougher tool to distinguish, classify, or retrieve shapes from a restricted population of surfaces. An example of such a situation is the subject of this Chapter, our second thesis objective: the classification and retrieval of human poses and motions.

Furthermore, the articulation of the human body enables it to adopt a great variety of poses with very small changes to the intrinsic geometry of the surface that models it. In flexing an arm or a leg we mostly see small intrinsic changes due to the bulging and stretching of muscles, but the net result in terms of the extrinsic geometry of the body can be substantial. Small changes in the intrinsic geometry may even lead to apparent changes in the genus of the human figure through topological noise when, for instance, hands are clasped or feet and legs are crossed. This points to the unsuitability of *intrinsic* approaches which are focused on the metric relations (lengths of curves, angles, and areas) on the surface itself independently of the embedding into the ambient space.

In the analysis and retrieval of human actions, we must work with sequences of a hundred human poses, and each pose is represented by a triangulated surface containing thousands or tens of thousands of vertices. This computational complexity is nevertheless offset by the fact that human poses are modeled by a rather restricted population of surfaces. Examination of the databases led us to formulate the hypothesis that a human pose is nearly characterized by its convex hull. The intuition is that if you enclose someone in a tight, perfectly elastic sheet, the different poses of this person will still be distinguishable, or mostly so (see Figure 4.1). In considering human body motion, where there is a sequence of poses, the probability of recognition of the action from the associated sequence of convex hulls should be even greater,

or so the intuition goes.

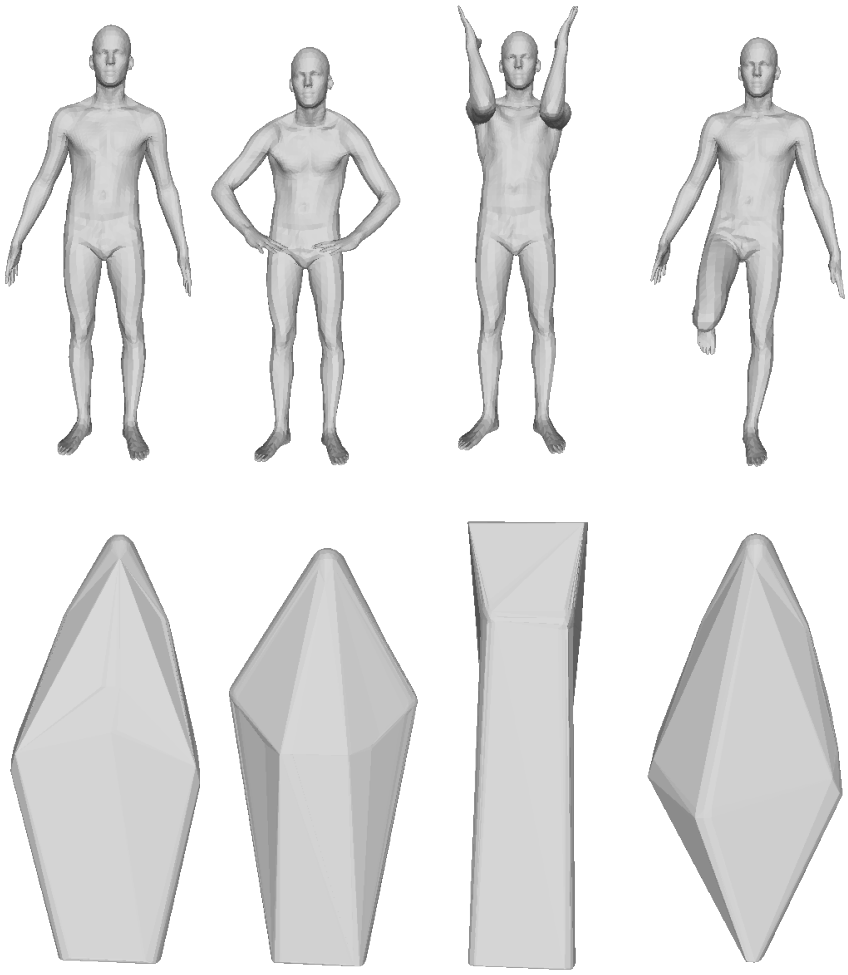


Figure 4.1 Four human poses from the FAUST dataset along with their corresponding convex hulls.

This convexity hypothesis led to the idea of considering two of the most basic notions in convex geometry, the convex hull, and the surface area measure or Extended Gaussian Image (EGI), and molding them into three sequences of numerical surface descriptors that are invariant under Euclidean transformations. We do this by first encoding the information of the convex hull in the breadth function, which measures the length of the projection of the surface onto each line passing through the origin, and encoding the information of the EGI in the weighted area function, which for each direction measures the weighted area of the projection of the surface onto the plane perpendicular to it (see Section 4.3 for details). These functions only depend on symmetrizations of the convex hull and EGI (Proposition 4.3.2 and Theorem 4.3.5), but are supplementary (i.e., two non-convex surfaces with the same symmetrized convex hull are not likely to have the same symmetrized EGI) and lend themselves nicely to Fourier analysis. Our three sequences of numerical shape descriptors are obtained as the L_2 norms and inner products of the projections of these functions onto the space of spherical harmonics of order

k. In geophysics terminology, these are the power spectra and the cross power spectrum of our two functions (Kaula, 1967; Lowes, 1974) introduced in the context of shape matching by (Kazhdan *et al.*, 2003).

The main concern of this chapter is the problem of analyzing human motion and our numerical descriptors conveniently allow us to reformulate it as a problem of comparing polygonal curves in $\mathbb{R}^n$. In this familiar setting we make use of Dynamic Time Warping (DTW) to compare the curves obtained from the CVSSP3D real and synthetic datasets (Starck & Hilton, 2007).

We did not pay close attention to the effect that noisy data could have on our methods and to the interesting problem of how to make them more robust, but we did test them against the relatively noisy CVSSP3D real dataset (see Table 4.3) and remarked that a slight modification to our breadth function to make it more robust yielded good results.

Overall, the contributions of this chapter can be summarized as follows.

- We present a novel set of descriptors invariant under parametrization, Euclidean transformations, and scaling.
- We formulate the problem of comparing sequences of 3D human surfaces as a problem of comparing curves in $\mathbb{R}^n$. Dynamic Time Warping is proposed for the temporal alignment of these curves.
- The method shows promising results for 3D pose and 3D motion retrieval tasks in several datasets. The results are promising and validate our hypothesis that the analysis of human action can be in good measure reduced to the analysis of sequences of convex hulls of human poses. The experimental results show that our method can be implemented in a computationally efficient way due to its simple formulation.

4.2 Related Work in 3D shape sequence retrieval

After the pioneer works in 3D human shape sequence classification made thanks to the popular Xmas dataset (Weinland *et al.*, 2006), a few works extended a selection of 3D shape descriptors to human motion retrieval. We refer to Chapter 2, section 1 to a broader view of available descriptors in literature. The descriptors that were applied to human motion retrieval, were first Shape Distribution, Spin Image, and Spherical Harmonics in (Huang *et al.*, 2010), but they are not necessarily adapted to the particular geometry of the human body. Slama *et al.* (Slama *et al.*, 2014) then used the Extremal Human Curve (EHC), a set of 10 curves that connect the extremal points of the 3D human surface. They propose a geometric approach for comparing those shapes, by exploiting the fact that curves can be parameterized canonically and thus can be compared naturally. However, the need for the detection of extremal points makes this approach sensitive to noise and to the low quality of the meshes. In addition, the comparison between all pairs of curves increases the computational cost. Another interesting approach is presented by Luo *et al.* (Luo *et al.*, 2016), where they compute a Spatio-temporal graph of 3D Human motion. However, this approach also suffers from being time-consuming and needs the same parameterization along a dataset to perform well. In (Veinidis *et al.*, 2019) six static shape descriptors are extracted from each mesh of the human sequence and DTW is used as a similarity measure, before proposing to add other information like centroid position and speed.

However, some descriptors used in this approach require pose normalization against rotations for each mesh per frame.

4.3 Projection-based classification of surfaces

4.3.1 The breadth representation

As we mentioned in the introduction, the guiding idea of this chapter is that human poses seem to be determined to a great extent by their convex hulls (see Figure 4.1). In order to quantify and test this hypothesis, we consider the support and breadth functions of the triangulated surfaces that model the human form.

Definition 4.3.1. *The support function of a set $S \subset \mathbb{R}^n$ evaluated at the unit vector $u \in S^{n-1}$ is the quantity*

$$h(S;u) := \sup_{x \in S} u \cdot x.$$

The breadth of the set $S \subset \mathbb{R}^n$ in the direction given by the unit vector $u \in S^{n-1}$ is the quantity

$$b(S;u) := h(S;u) + h(S,-u) = \sup_{x \in S} u \cdot x - \inf_{x \in S} u \cdot x \ .$$

Geometrically speaking, the breadth of a path-connected set in a direction u is simply the length of the orthogonal projection of the set onto a line parallel to u. As the following classic result shows, the support function is a way to encode the convex hull.

Proposition 4.3.2. *Two sets $S_1, S_2 \subset \mathbb{R}^n$ have the same support function if and only if their convex hulls are equal. Their breadth functions are the same if and only the convex hulls of the sets $S_1 - S_1$ and $S_2 - S_2$ are equal.*

Proof. The convex hull of a set is the intersection of all half-spaces that contain it. From the definition of the support function, for each unit vector u, the half-space

$$H(S;u) := \{x \in \mathbb{R}^n : u \cdot x \leq h(S;u)\}$$

contains S and is minimal in the sense that it is the unique half-space that contains S and is contained in $H(S;u)$. From this perspective, the support function is just a way to encode the set of minimal half-spaces, and thus the set of all half-spaces, that contain S. It follows that the support function of a set characterizes its convex hull.

From the linearity of the functions $x \mapsto u \cdot x$ and the definition of support function, we have that if A and B are two subsets of $\mathbb{R}^n$, and λ_1 and λ_2 are two positive numbers, then

$$h(\lambda_1 A + \lambda_2 B;u) = \lambda_1 h(A;u) + \lambda_2 h(B;u) \text{ and } h(-A,u) = h(A;-u).$$

From this we conclude that the breadth function of a set S is also the support function of $S - S$:

$$b(S;u) = h(S;u) + h(S;-u) = h(S-S;u).$$

Since the convex hull of a set is characterized by its support function, we conclude that the convex hulls of the sets $S_1 - S_1$ are the same if and only if the breadth functions of S_1 and S_2 are equal.

Unlike the breadth function, the support function is not invariant under translations. This can be fixed by moving the center of mass to the origin. Generally speaking, there is less loss of information when working with the support function than with the breadth function, and this should come up in comparing surfaces that have a central symmetry to those that do not. However, for comparing human figures this did not seem to be the case and we made the choice to work with the breadth function to keep within a geometric tomography framework of studying human shapes through their projections onto lines and planes.

Using that triangles are convex and that the functions $x \mapsto u \cdot x$ ($u \in S^2$) are linear, the breadth of a triangulated surface $M \subset \mathbb{R}^3$ can be easily computed from just the knowledge of its vertex points $x_1, \ldots, x_N$:

$$b(M;u) := \max_{1 \le i \le N} u \cdot x_i - \min_{1 \le i \le N} u \cdot x_i \ .$$

4.3.2 Area representation

Another classical descriptor of convex bodies and surfaces is the *surface area measure* or, as is better known in computer vision, the *extended Gaussian image* (EGI). This is the push-forward of the two-dimensional Hausdorff measure of the surface onto the unit sphere under the Gauss map. For a triangulated surface, we can give a more pedestrian equivalent formulation:

Definition 4.3.3. *Given an oriented triangulated surface $M \subset \mathbb{R}^3$ formed by a union of triangles $T_1 \ldots, T_m$, its extended Gaussian image is the measure on the unit sphere*

$$\mu_M := \sum_{i=1}^{m} area(T_i)\, \delta_{n_i},$$

where n_i is the unit vector perpendicular to T_i in the sense defined by the orientation of the surface and δ_{n_i} is the delta measure concentrated at n_i.

There are a number of ways to extract feature vectors from the EGI of a surface. We can, for instance, manufacture them from the moments or the Fourier transform of this measure, but in this work we chose a more intuitive descriptor: the *weighted area function.*

Definition 4.3.4. *Given an oriented triangulated surface $M \subset \mathbb{R}^3$ formed by a union of triangles $T_1 \ldots, T_m$, its weighted area function is the function on the unit sphere defined by*

$$\mathscr{A}(M;u) := \sum_{i=1}^{m} |u \cdot n_i|\, area(T_i),$$

where n_i is a unit vector perpendicular to the triangle T_i.

The quantity $\mathscr{A}(M;u)$ is the weighted area of the projection of M onto the plane orthogonal to u. By *weighted area* we mean that if k different portions of a surface project onto the same piece of plane, the area of this piece is multiplied by k.

Besides being invariant under reparametrizations and translations, the weighted area function is easy to grasp geometrically and very quickly computed. It's relation to the EGI of the surface follows directly from the definitions:

$$\mathscr{A}(M;u) = \int_{S^2} |u \cdot n|\, d\,\mu_M.$$

This expression immediately implies that surfaces with the same EGI are indistinguishable by the weighted areas of their projections. Moreover, because the functions $x \mapsto |u \cdot x|$ $(u \in S^2)$ are even, we only see the *even part* of the measure μ_M,

$$\mu_M^e = \frac{1}{2}\sum_{i=1}^{m} \text{area}(T_i)(\delta_{n_i} + \delta_{-n_i}).$$

It follows that if the even parts of the surface area measures of two oriented surfaces are the same, then their weighted area functions are identical. This is all: by a theorem of Choquet ((Choquet, 1969, p. 53)), finite linear combinations of the functions $x \mapsto |u \cdot x|$ $(u \in S^2)$ are dense in the space of even continuous functions on the sphere, and hence if the integrals of all functions of this form with respect to two even measures are the same, the measures must be the same. We summarize:

Theorem 4.3.5. *Two oriented triangulated surfaces $M_1, M_2 \subset \mathbb{R}^3$ are indistinguishable by the weighted areas of their projections if and only if the even parts of their extended Gaussian images are the same.*

In order to use the weighted area function as a descriptor it is important to understand that if we decompose a surface into a finite or countable number of pieces each of which has a computable area, translating these pieces or flipping them around the origin, and then recomposing them again will give a new surface whose projection onto any plane has the same weighted area as that of the original surface. For instance, if we wish to make use of this technique to classify poses of a human figure it is useful to keep in mind the following rule of thumb: if we approximate and decompose the human body as the union of a number of boxes and then these boxes are moved by pure translation and re-glued into a different pose, the method will not effectively distinguish the old and the new poses. An important example is a person standing up with the arms by his/her side and the same person standing up with the arms straight up over his/her head (see Fig. 4.2).

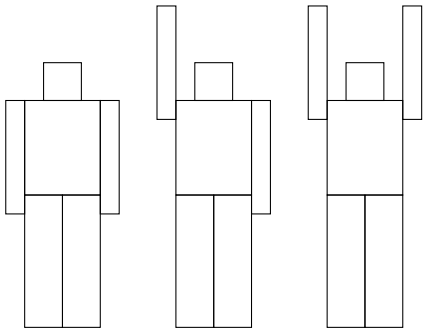


Figure 4.2 Different poses with the same weighted area function, but with different breadth functions.

Because of this “cut-translate-and-paste" invariance, the weighted area may not seem to be as good a descriptor as the breadth, and indeed, that is what our results confirm (see Table 4.1), but it is supplementary information and can be quite discerning in its own right. The weighted area allows us to distinguish some non-convex surfaces that have the same convex hull or breadth

function, and although it is possible for two different non-convex surfaces to have the same convex hull and EGI—and, a fortiori, the same breadth and weighted area functions—without being translates (see Figure 4.3 for a simple two-dimensional example of these phenomena), that does not seem to happen to any significant degree in the restricted population of human poses. Nevertheless, the real advantage of considering simultaneously the breadth and weighted area functions will become clearer when we tackle the problem of extracting Euclidean invariants from these functions.

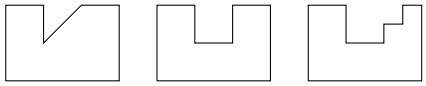


Figure 4.3 The first two forms have the same convex hull and different weighted area functions, while the last two forms have the same convex hull and EGI.

4.3.3 Euclidean and shape invariants

In many applications it is not enough to be able to distinguish or classify surfaces up to reparametrizations and translations. Often we need to do so up to Euclidean transformations or up to similarities. In this section we describe a simple method to extract sequences of Euclidean and shape invariants from the area and breadth function of a surface.

Notice that if $M \subset \mathbb{R}^3$ is a surface and R is a 3×3 orthogonal matrix, then

$$\mathscr{A}(RM;u) = \mathscr{A}(M;R^{-1}u) \text{ and } b(RM;u) = b(M,R^{-1}u)$$

for every unit vector u. In other words, the assignments $M \mapsto \mathscr{A}(M;\cdot)$ and $M \mapsto b(M;\cdot)$ are $O(3)$-equivariant maps between the space of surfaces and the space $L_2(S^2)$ of square-integrable functions on the sphere provided with the usual left $O(3)$-action $(R,f) \mapsto f \circ R^{-1}$. The classic theory of spherical harmonics (see Lecture 11 of (Arnold, 2004) for a particularly simple description) tells us that this space decomposes into the direct sum

$$L_2(S^2) = \mathbb{R} \oplus V_1 \oplus V_2 \oplus \cdots,$$

where V_k is the $(2k+1)$-dimensional space of spherical harmonics of order k (i.e., the restriction to the sphere of homogeneous harmonic polynomials of order k in $\mathbb{R}^3$). These subspaces are invariant under the action of the orthogonal group and are mutually orthogonal. It follows that if f is a square integrable function on the sphere, we can decompose $f = f_0 + f_1 + f_2 + \ldots$ with $f_k \in V_k$, and that the L_2 norm of each component f_k, defined by

$$||f_k||_2^2 := \frac{1}{4\pi} \int_{S^2} f_k(u)^2 \, d\Omega,$$

is invariant under the orthogonal group. Notice that if the function f is an even function, all the odd terms, f_{2k+1}, $k \geq 0$, are zero. This method to extract rotation invariants from spherical functions is classical (see, for instance, (Weyl, 1946)) and is widely used in geophysics ((Kaula, 1967; Lowes, 1974)), but in the context of computer science it seems to have been introduced

in (Kazhdan *et al.*, 2003), where the term *energy representation* of f is used for the sequence $k \mapsto ||f_k||_2$.

Applying this idea to the area and breadth functions of a surface M we obtain two sequences of invariants

$$\alpha_k(M) := ||\mathscr{A}_{2k}(M;\cdot)||_2 \text{ and } \beta_k(M) := ||b_{2k}(M;\cdot)||_2.$$

To this we add the sequence $\gamma_k(M)$ consisting of the inner products of $\mathscr{A}_{2k}(M;\cdot)$ and $b_{2k}(M;\cdot)$:

$$\langle \mathscr{A}_{2k}(M;\cdot), b_{2k}(M;\cdot)\rangle_2 = \frac{1}{4\pi}\int_{S^2} \mathscr{A}_{2k}(M;u)b_{2k}(M;u)\,d\Omega,$$

which is also a Euclidean invariant of the surface M.

Using the equality

$$||f+g||_2^2 = ||f||_2^2 + 2\langle f,g\rangle_2 + ||g||_2^2,$$

we have that

$$\gamma_k(M) = \frac{1}{2}\left(||\mathscr{A}_{2k}(M,\cdot) + b_{2k}(M,\cdot)||_2^2 - \alpha_k^2(M) - \beta_k^2(M)\right).$$

It is not clear what is the geometric meaning of most of these invariants, but by the Cauchy-Crofton formula $\alpha_0(M)$ is simply one-fourth the area of M, while $\beta_0(M)$ is $(1/2\pi)$ times the integral of the mean curvature of M, *provided the surface is convex* (see Chapter 14 in (Santaló, 1976)).

In practice we only know the values of the functions $\mathscr{A}(M;\cdot)$ and $b(M;\cdot)$ on a finite set of grid nodes. Through the use of FFT and cubature formulas it is possible to numerically compute the invariants $\alpha_k(M)$, $\beta_k(M)$, and $\gamma_k(M)$ for $0 \le k \le l$, where $16(l+1)^2$ is the number of nodes in our grid (see (Wieczorek & Meschede, 2018, pp. 2580–2581)). Thus, the $l \times 3$ matrix

$$\mathscr{E}_l(M) := \begin{pmatrix} \alpha_0(M) & \beta_0(M) & \gamma_0(M) \\ \vdots & \vdots & \vdots \\ \alpha_l(M) & \beta_l(M) & \gamma_l(M) \end{pmatrix},$$

which will be our basic Euclidean-invariant representation of the surface M, can be effectively computed from the values of the area and breadth functions of M over a uniform sample of $16(l+1)^2$ points on the sphere.

To end this section we briefly discuss how to extend these Euclidean invariants to shape or similarity invariants, where we allow for dilations as well as rotations and translations. To do this we note that if λ is a positive real number, then

$$\mathscr{A}(\lambda M;u) = \lambda^2 \mathscr{A}(M;u) \text{ and } b(\lambda M;u) = \lambda b(M;u).$$

It follows that

$$\alpha_k(\lambda M) = \lambda^2 \alpha_k(M),\ \beta_k(\lambda M) = \lambda \beta_k(M),$$
$$\gamma_k(\lambda M) = \lambda^3 \gamma_k(M).$$

We can get rid of the dilation factor in a number of ways. For instance, for each $k \ge 0$, the quantities

$$\alpha'_k(M) := \frac{\alpha_k(M)}{||\mathscr{A}(M,;\cdot)||_2} \text{ and } \beta'_k(M) := \frac{\beta_k(M)}{||b(M;\cdot)||_2}$$

are shape invariants of M, as is

$$\gamma'_k(M) := \left\| \frac{\mathscr{A}_{2k}(M,\cdot)}{\|\mathscr{A}(M,\cdot)\|_2} + \frac{b_{2k}(M,\cdot)}{\|b(M,\cdot)\|_2} \right\|_2 .$$

As the reader can see, $\gamma'_k(M)$ does not resemble $\gamma_k(M)$ as much as the primed versions of $\alpha_k(M)$ and $\beta_k(M)$ resemble their original versions, but because of the numerical issues we will now discuss, it will be useful for us to have only non-negative shape invariants.

4.3.4 Numerical considerations

Since the spherical harmonic expansions of the functions $\mathscr{A}(M;\cdot)$ and $b(M;\cdot)$ converge, it follows from Parseval's identity that the invariants $\alpha_k(M)$, $\beta_k(M)$, and $\gamma_k(M)$ tend to zero. They would even decay exponentially if the functions were smooth (see (Livermore, 2012, p. 1151) for a quick proof). In fact, neither function is smooth: the first is a finite convex sum of the non-smooth functions $u \mapsto |u \cdot n_i|$, and the second is support function of a polytope, namely the convex hull of the differences of all pairs of vertices in the triangulated surface. However, experimentally (and perhaps due to the great number and small size of the triangles in our triangulated surfaces) the batch of invariants we computed does exhibit exponential decay. Therefore, the last rows of our basic Euclidean representation

$$\mathscr{E}_l(M) := \begin{pmatrix} \alpha_0(M) & \beta_0(M) & \gamma_0(M) \\ \vdots & \vdots & \vdots \\ \alpha_l(M) & \beta_l(M) & \gamma_l(M) \end{pmatrix},$$

will be nearly all zero for even relatively small values of l. We would prefer to deal with invariants that decay at a slower rate to give some, but not too much, weight to higher harmonics. To be precise, what worked for us was a $t \mapsto 1/t$ decay. To achieve this we change $\alpha'_k(M)$ for

$$\alpha^s_k(M) := \begin{cases} -\ln(\alpha'_k(M))^{-1} & \text{if } \alpha'_k(M) > 0, \\ 0 & \text{if } \alpha'_k(M) = 0. \end{cases}$$

Similarly, we change $\beta'_k(M)$ for

$$\beta^s_k(M) := \begin{cases} -\ln(\beta'_k(M))^{-1} & \text{if } \beta'_k(M) > 0, \\ 0 & \text{if } \beta'_k(M) = 0, \end{cases}$$

and, lastly, we change $\gamma'_k(M)$ for

$$\gamma^s_k(M) := \begin{cases} -\ln(\gamma'_k(M))^{-1} & \text{if } \gamma'_k(M) > 0, \\ 0 & \text{if } \gamma'_k(M) = 0. \end{cases}$$

From now on we will be working with the modified shape invariant

$$\mathscr{E}^s_l(M) := \begin{pmatrix} \alpha^s_0(M) & \beta^s_0(M) & \gamma^s_0(M) \\ \vdots & \vdots & \vdots \\ \alpha^s_l(M) & \beta^s_l(M) & \gamma^s_l(M) \end{pmatrix}.$$

4.3.5 Representation of surfaces and surface evolution

The final aim of all the preceding mathematics is to represent surfaces as points and discrete surface motions as polygonal curves in a suitable feature vector space. We consider two types of representation, both of which are independent of the parametrization of the surface: a translation-invariant representation and a shape-invariant representation.

To obtain a translation-invariant representation of a surface M we take a regular sample of n latitude angles, along with a regular sample of n longitude angles of the sphere. We combine them to obtain a spherical grid with n^2 nodes $u_1, \ldots, u_{n^2}$ and represent M by one of the following vectors:

1. The *breadths* feature vector
$$(b(M;u_1), \ldots, b(M;u_{n^2})) \in \mathbb{R}^{n^2}.$$
2. The *areas* feature vector
$$(\mathscr{A}(M;u_1), \ldots, \mathscr{A}(M;u_{n^2})) \in \mathbb{R}^{n^2}.$$
3. The *areas & breadths* feature vector which is obtained by joining the previous two:
$$(\mathscr{A}(M;u_1), \ldots, \mathscr{A}(M;u_{n^2}), b(M;u_1), \ldots, b(M;u_{n^2})).$$

To obtain a shape-invariant representation of M we take a similar spherical grid of $16n^2$ nodes and use the values of $\mathscr{A}(M;\cdot)$ and $b(M;\cdot)$ on these nodes to compute the shape-invariant matrix $\mathscr{E}^s_{n-1}$. Since we wish to understand how discerning the energies of the breadth and the weighted area functions are, we shall also consider the first two columns of $\mathscr{E}^s_{n-1}$ separately. This gives us three shape-invariant feature vectors:

4. The *area spectrum:*
$$(\alpha^s_0(M), \ldots, \alpha^s_{n-1}(M)).$$
5. The *breadth spectrum:*
$$(\beta^s_0(M), \ldots, \beta^s_{n-1}(M)).$$
6. The shape invariant $\mathscr{E}^s_{n-1}$.

In this chapter we will set $n = 8$ and hence when dealing with translation-invariant feature vectors we will be working either in $\mathbb{R}^{64}$ or $\mathbb{R}^{128}$, and when dealing with shape-invariant feature vectors we will be working either in $\mathbb{R}^8$ or $\mathbb{R}^{24}$. In all cases we will be using the standard Euclidean metric in these spaces to compare surfaces through their associated vectors.

In order to analyze human motion, we need to find a representation for a sequence of surfaces with timestamps, $(M_0, t_0), \ldots, (M_p, t_p)$. Using any one of the six feature vectors described above we associate to this sequence a parametrized polygonal curve in a feature vector space: if $f(M)$ denotes our feature vector, we construct the polygonal curve whose vertices are $\mathbf{x}_j := f(M_j)$, and for which the parametrization in each segment $\mathbf{x}_j\mathbf{x}_{j+1}$ is given by

$$t \longmapsto \frac{t - t_{j+1}}{t_j - t_{j+1}}\,\mathbf{x}_j + \frac{t - t_j}{t_{j+1} - t_j}\,\mathbf{x}_{j+1}$$

for $t_j \leq t \leq t_{j+1}$ and $0 \leq j \leq p-1$.

By this procedure the problem of comparing two human motions, or any other two discrete surface motions, is then reduced to that of choosing a suitable feature vector and comparing the two parametrized polygonal curves associated to the motions.

4.4 Experiments

4.4.1 Evaluation setup

We test the usefulness of the proposed descriptors in two applications: static 3D human pose and 3D human motion retrieval.

Metric evaluation. We use three evaluation measures. For all measures a high score implies better results.

1. **Nearest neighbor (NN):** It equals one if the nearest neighbor is of the same class of the query, 0 otherwise. This statistic provides an indication of how well a nearest neighbor classifier would perform.

2. **First-tier (FT), Second-tier (ST):** the percentage of models in the query's class C that appear within the top K matches, K depending on query's class size. For a class with $|C|$ members, $K = |C| - 1$ for the first tier, and $K = 2 \times (|C| - 1)$ for the second tier.

The score displayed in evaluation tables are the mean scores computed over the dataset.

4.4.2 Datasets

We work in this chapter with the following datasets :

- The FAUST dataset, where we use the 10 training poses for pose retrieval
- The CVSSP3D artificial dataset as a dataset for motion retrieval
- The CVSSP3D real dataset as a dataset for motion retrieval

4.4.3 Static pose retrieval on the FAUST dataset

Each pose of a dataset is considered as a query belonging to some class. We compute the Euclidean distance between the query pose descriptors and each pose in the dataset (Figure 4.4).

Comparison with state-of-the-art. In order to evaluate our descriptor against available methods in the literature, we compare to the following approaches:

1. Skinned Multi-Person Linear model (SMPL) pose representation. The SMPL body model (Loper *et al.*, 2015) is composed of three parts: a template mesh, a pose vector, and a shape vector. The shape vector represents the (non-rigid) deformation of the template to the shape of the given human body. The pose information of a skeletal joint is the relative rotation of the joint of the skeleton compared to its parent joint, and is stored either as the rotation matrix or as axis-angle representation. We convert each joint

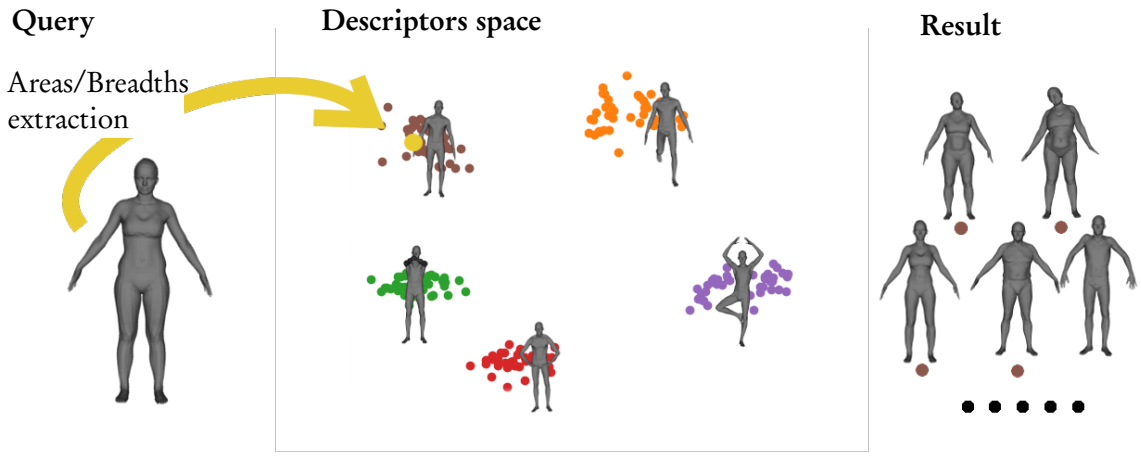


Figure 4.4 Overview of our pose retrieval approach: We first compute the descriptors (Areas/Breadths or Areas & Breadths) of all shapes in the database. Given a query shape, we compute its corresponding descriptor and collect the closest shapes in the descriptor space.

rotation to quaternion representation as in (Zhou *et al.*, 2020a; Aumentado-Armstrong *et al.*, 2019) and measure the distance between unit quaternions by $d(q,q') = 1 - |q.q'|$. The SMPL body pose vector contains the pose information of 52 joints, and the rotation of the central joint accounts for the global rotation of the shape. The representation is a point in $(\mathbb{R}^4)^{51} = \mathbb{R}^{204}$. Due to the construction of the pose vector, this descriptor is rotation invariant. However, this method is time consuming compared to ours because of the needed fitting operation to the mesh.

2. Aumentado-Armstrong *et al.* (Aumentado-Armstrong *et al.*, 2019) propose Geometrically Disentangled VAE (GDVAE), a point cloud variational autoencoder which is trained to disantengle the intrinsic and extrinsic informations of a given shape in the latent space. The authors propose the intrinsic and extrinsic latent vectors for human shape representation. We used the FAUST meshes as input of their available trained network, gathered their extrinsic latent vectors (belonging to $\mathbb{R}^{12}$), and used them for human pose retrieval. Although the procedure is parametrization invariant by nature (the networks takes a cloud of points as input), the training uses the mesh Laplacian as ground truth information, and this means a constant parameterization along the training set. The network is trained on the SURREAL dataset (Varol *et al.*, 2017) in such a way as to be rotation invariant.

3. Zhou *et al.* (Zhou *et al.*, 2020a) propose a mesh autoencoder based on the Neural3DMM (Bouritsas *et al.*, 2019) graph neural network structure. As in the case of GDVAE, this autoencoder disantengles shape and pose in latent space. The network requires that all input meshes have the same parameterization. We apply the FAUST meshes on their available network trained on the AMASS dataset, and use the pose latent vector (belonging to $\mathbb{R}^{112}$) as a descriptor for comparison. Since the input of the network are the coordinates of the vertices, the approach is not rotation invariant.

Representation	NN	FT	ST	Computation time
GDVAE (Aumentado-Armstrong *et al.*, 2019)	60	38.0	54.2	190ms
Zhou *et al.* (Zhou *et al.*, 2020a)	82	69.2	83.4	30.7ms
SMPL pose vector	80	**84.4**	**95.2**	$\approx 5min$
Areas	62	50.0	67.2	**4.1ms**
Breadths	83	63.1	76.6	13.2ms
Areas & Breadths	**86**	67.9	80.9	17.2ms

Table 4.1 Results for pose retrieval and computation time for feature extraction for FAUST dataset. The computations were performed with NumPy routines on a Intel(R) Core(TM) i5-7600K 3.8GHz CPU, with 8GB of RAM available, except for SMPL, for which the given method needed the use of a GPU.

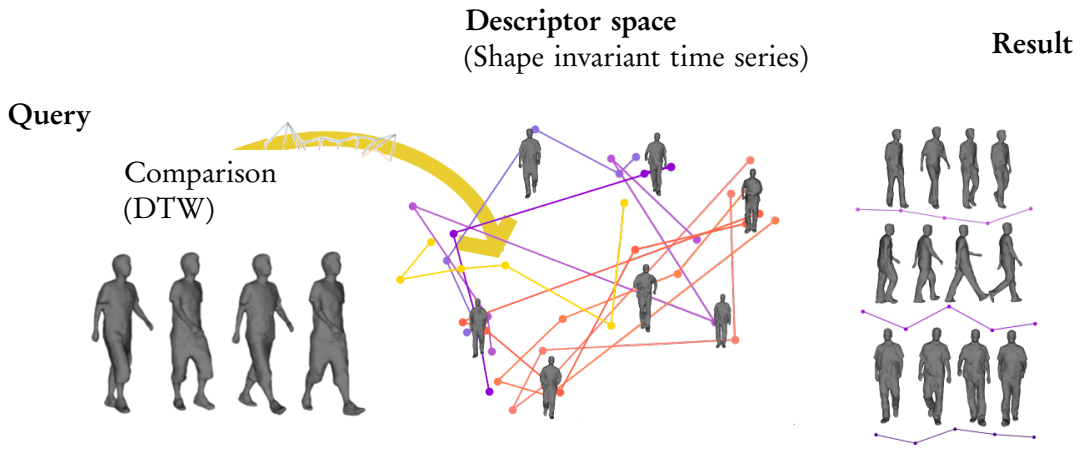


Figure 4.5 Overview of our motion retrieval approach. We first compute the time series of descriptors (areas/breadth spectra or shape invariant) of all motions in the database. Given a query shape, we compute its corresponding time series and compare it against the time series of the database in the descriptor space using dynamic time warping. We then collect the closest motions given this similarity

Table 4.1 displays the results obtained for the Areas, Breadths, and Areas & Breaths descriptors. The results for the Breadths descriptor is of particular interest as it is here where we see the high correlation between poses and their (symmetrized) convex hull, which validates our main hypothesis. In fact, Breadth by itself outperforms all previous methods in the NN criterion. When complemented by areas, the performance improves by 3%. The results also show that the SMPL pose vector performs much better for the FT and ST metrics. This result can be explained by the fact that SMPL has been designed specifically for human shapes. In addition, the SMPL fitting method used here requires a dataset of meshes registered to a template. The last column of Table 4.1 shows that our approach is faster than all the methods. It shows also that the computation time of SMPL descriptor is very high.

4.4.4 3D Human Motion retrieval on CVSSP3D artificial dataset

Each mesh sequence of a dataset is considered as a query belonging to some class. We compute the DTW similarity between the query mesh sequence and each mesh sequence in the dataset

(Figure 4.5).

Comparison with state-of-the-art. An extensive comparison has been made in (Veinidis *et al.*, 2019) to evaluate a bench of descriptors for human motion retrieval. The polygonal curves of those descriptors are filtered with a temporal filtering approach(a mean filter is applied along a temporal window of size K). Finally, the dynamic time warping distance is used for comparing the resulting curves. We compare our invariant descriptors (breadth and area spectrum, shape invariant) to the euclidean and parameterization invariant features presented in (Veinidis *et al.*, 2019), which are:

1. Shape Distribution (Osada *et al.*, 2002)(Veinidis *et al.*, 2019) is a 3D descriptor based on pairwise distances. All pairwise distances of a given shape are computed, and the resulting descriptor is an histogram of the obtained distances.

2. Spin Images (Johnson, 1997)(Veinidis *et al.*, 2019) is a 3D shape descriptor based on local features. For each point of a shape, a view from the point (the spin image) is computed, which takes the form of a 2D histogram. The resulting descriptor is the sum of all spin images.

3. The pretrained GDVAE on SURREAL is applied directly on the dataset. It does not need any supplementary work since the network (PointNet) is parameterization invariant.

4. The Neural3DMM autoencoder from (Zhou *et al.*, 2020a) needs to be specifically trained on the CVSSP3D artificial dataset, since the network is set to specific mesh parameterization and alignment. In order to be fair to the other methods that were not trained on the dataset, we apply a cross identity validation to compute the score. For each individual, we remove its motions from the training dataset. We then compute the retrieval scores for the individual motions using the trained pose representation. The training setting is exactly the same as in (Zhou *et al.*, 2020a).

Representation	NN	FT	ST
Shape Distribution (Osada *et al.*, 2002)(Veinidis *et al.*, 2019)	92.1	88.9	97.2
Spin Images (Johnson, 1997)(Veinidis *et al.*, 2019)	100	87.1	94.1
GDVAE (Aumentado-Armstrong *et al.*, 2019)	100	97.6	98.8
Zhou *et al.* (Zhou *et al.*, 2020a)	100	99.6	99.6
Area spectrum	81.6	56.6	68.2
Breadth spectrum	100	**99.8**	100
Shape invariant $\mathscr{E}_7^s$	82.1	56.8	68.5

Table 4.2 CVSSP3D artificial dataset results for motion retrieval using our shape-invariant representations. The results of Shape Distributions and Spin Images are reported from (Veinidis *et al.*, 2019).

We report our results on CVSSP3D artificial dataset in Table 4.2. The window sizes for temporal filtering applied to Shape Distribution and Spin Images are 9 and 8 respectively as in (Veinidis *et al.*, 2019). Our method did not require temporal filtering. We observe that the breadth spectrum has the best performance, near 100%, in all criteria.

4.4.5 3D Human Motion retrieval on CVSSP3D real dataset

The CVSSP3D real dataset differs significantly from the artificial human motion dataset because of the relatively noisy data (see Figure 4.6) and the various kinds of loose-fitting clothes in some of the models (see Figure 4.6 and Figure 4.7). This raises the problem of making our

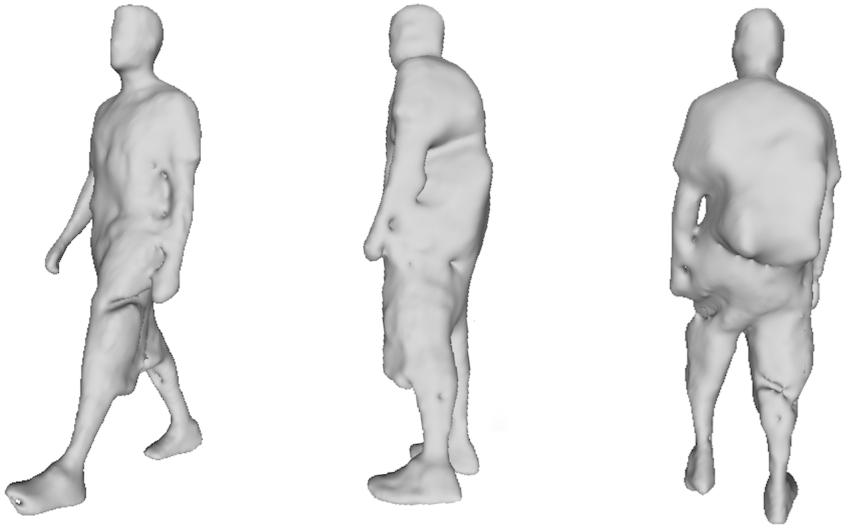


Figure 4.6 Examples of artifacts in the CVSSP3D real dataset.

descriptors more robust. While a thorough study of this question will be left for a future publication, two conceptually simple and easily implemented modifications to our method can have a significant impact.

The λ-percentile breadth function. The breadth function is particularly sensitive to outliers: the maximum or the minimum value of the function $x \mapsto u \cdot x$ can change significantly with a single noisy vertex x. To make this descriptor more robust we make a simple change to the support function of a finite set:

Definition 4.4.1. *Given a finite set $S \subset \mathbb{R}^n$ and a parameter λ, $0 < \lambda \leq 100$, we define the λ-percentile support function of S as the function $h_\lambda(S,\cdot)$ that assigns to a unit vector $u \in S^{n-1}$ the λ-th percentile of the values $\{u \cdot x : x \in S\}$. The λ-percentile breadth function of S is given by*

$$b_\lambda(M;u) = h_\lambda(S;u) + h_\lambda(S;-u).$$

Defined in terms of the vertices of a triangulation, $b_\lambda(M;\cdot)$ is *not* invariant under re-triangulations of the surface for $\lambda < 100$. It is only approximately so if the mesh is fine enough and the sizes and shapes of all triangles are comparable. Nevertheless, it is invariant under translations of M and satisfies the equivariance condition

$$b_\lambda(RM;u) = b_\lambda(M;R^{-1}u).$$

Provided we understand the conditions on the meshes of the surfaces we are working with, we can use $b_\lambda(M;\cdot)$ as a substitute of the breadth function in the construction of shape invariants detailed in Section 4.3.3. We experimented with various values for λ and settled on the classic third quartile $\lambda = 75$. We call the function $b_{75}(M;u)$ *Q-breadth.* The analogue of the shape-invariant $\mathscr{E}_8^s$ computed with the Q-breadth function instead of the breadth function will be called the *Q-shape invariant.*

Temporal filtering Our second trick consists in slightly changing the way we assign polygonal curves to sequences of surfaces with timestamps by making use of a special feature of our invariants. If we are given a sequence of surfaces we can consider their average breadth function and their average weighted area function, and then proceed with the construction of the feature vectors. Note that for the breadth spectrum, the area spectrum and the shape invariant this is not the same as averaging the feature vectors themselves (we tried that too: the results were not as good). This particularity of our representation allows us the possibility to perform a simple discrete convolution or temporal filtering on the data: given a sequence of surfaces with timestamps, $(M_0,t_0),\ldots,(M_p,t_p)$ and a number K, $0 < K < p$ we consider the timestamped averages of breadth and weighted area functions, which are both represented here by f to avoid redundancy,

$$\bar{f}_{t_i}(M;u) := \frac{1}{2K+1}\sum_{-K\le j\le K} f(M_{i+j};u),\ K \le i \le p-K.$$

With the sequence of timestamped averaged functions

$$\bar{f}_{t_K}(M;u),\ldots,\bar{f}_{t_{p-K}}(M;u)$$

we construct our timestamped feature vectors and the corresponding polygonal curve as described in Section 4.3.5. Note that this temporal filtering approach is slightly different from the one proposed in (Veinidis *et al.*, 2019) – our approach is using the specific structure of our descriptors. The results of our experiments and comparisons on the CVSSP3D real dataset are reported in Table 4.3. Again we report the results of Shape distances and Spin Images from (Veinidis *et al.*, 2019). We display in this table the used windows size for temporal filtering of each method. For this relatively noisy dataset, the table clearly shows the advantage of using the spectrum of the Q-breadth function and the Q-shape invariant.

The results in Table 4.3 show that the Q-shape invariant outperforms all other methods, including the deep learning method GDVAE whose performance drops significantly in the presence of noise. This can be explained by the noise-sensitivity of the spectrum of the Laplace-Beltrami Operator.

A remarkable difference between the results in Table 4.3 and those of Table 4.2 is that the first tier measure is quite low compared to the NN measure for all features. In order to give an idea of how the tier are distributed, a first tier query is illustrated in Table 4.7.

4.4.6 Computation times

Our methods were implemented using Numpy routines, with no other optimization. The computations were performed with NumPy routines on a Intel(R) Core(TM) i5-7600K 3.8GHz CPU, with 8GB of RAM available.

Repr.	K	NN	FT	ST
Shape Distribution (Osada *et al.*, 2002)	1	77.5	**51.6**	65.5
Spin Images (Johnson, 1997)	6	66.3	43.2	59.5
GDVAE (Aumentado-Armstrong *et al.*, 2019)	14	38.7	31.6	51.6
Area spectrum	14	67.5	47.0	63.2
Breadth spectrum	15	63.7	39.1	52.5
Q-breadth spectrum	5	80.0	44.8	59.5
Shape invariant $\mathscr{E}_7^s$	15	62.5	41.8	57.9
Q-shape invariant	4	**82.5**	51.3	**68.8**

Table 4.3 CVSSP3D real dataset results for motion retrieval using our shape-invariant representations and their Q-versions. The results of Shape Distributions and Spin Images are reported from (Veinidis *et al.*, 2019). The K value is the best window size for temporal filtering, and the displayed score are the corresponding best scores.

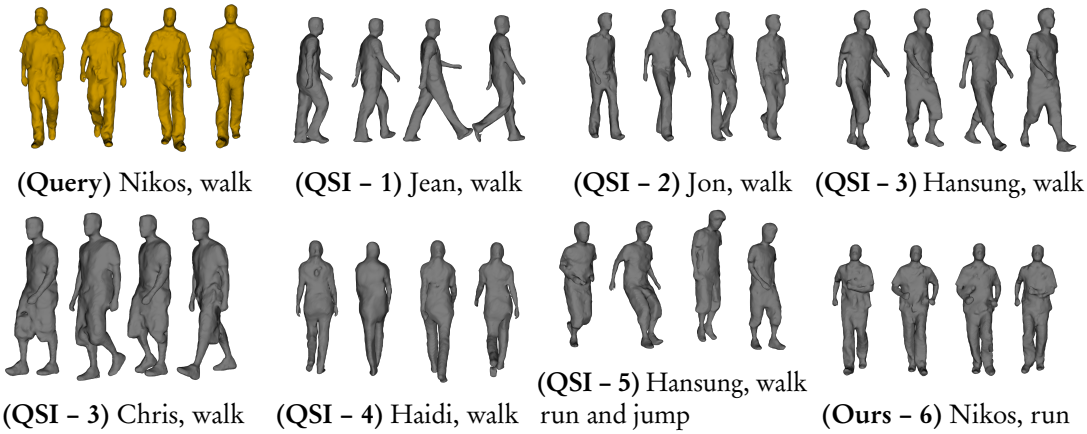


Figure 4.7 First tier of the query *Nikos, walk* for Q-shape invariant. The query is in yellow and the results are sorted by closeness to the query, using our Q-shape invariant descriptor.

In Table 4.1, last column, we present the computation times of each of the method applied to FAUST. For Zhou *et al.* (Zhou *et al.*, 2020a) and Aumentado-Armstrong *et al.* (Aumentado-Armstrong *et al.*, 2019), we used the implementations provided by the authors. For SMPL, we used the SMPL fitting pipeline proposed by the authors. In Table 4.4 we present the computation time of each method for the CVSSP3D datasets. For Zhou *et al.* (Zhou *et al.*, 2020a) and Aumentado-Armstrong *et al.* (Aumentado-Armstrong *et al.*, 2019) (GDVAE) we used the implementation provided by the authors. For Shape Distribution we use the hybrid Python-C implementation provided by Nenad Markuš [1]. For Spin Images, we used the C++ implementation provided by the PointCloud library [2]. We can see that our approach is the fastest on FAUST and CVSSP3D artificial datasets. We observe that the Q-shape invariant computation time is a bit slower than Shape Distribution for our approach in the real dataset – but the performance of our approach improves the NN criteria by 5%.

[1] https://nenadmarkus.com/p/shape-distributions
[2] https://pointclouds.org/documentation/classpcl_1_1_spin_image_estimation.html

Method	Real, 37800 vert.	Artif., 1290 vert.
Shape Dist.	79.1s*	61.2s*
Spin Image	3h54*	35.7s*
GDVAE	56.4s	2.08s
Shape invariant $\mathscr{E}_7^s$	**46s**	**1.7s**
Q-shape invariant	209s	/

Table 4.4 Mean computation time of polygonal curves extraction for different methods in the CVSSP3D datasets, along with the time corresponding to the polygonal curves in $\mathbb{R}^{24}$ using the Shape invariant $\mathscr{E}_7^s$, and the Q-shape invariant. We put the number of vertices for each dataset. Methods with an asterisk means that the implementation is not the official implementation provided by the authors.

4.5 Limitations and future work

4.5.1 Future work for human pose descriptors

Several avenues of future work are worth pursuing. We list some most promising directions below:

- It is theoretically possible to find a richer set of Euclidean invariants of convex bodies (Engel & Laasch, 2020; Kousholt & Schulte, 2021) using less harmonics. This would make the method more robust to noise.
- The noisy CVSSP3D real dataset has been a challenge for our descriptors. Some research should be spent on statistical analysis as in in (Poonawala *et al.*, 2002) to improve performance on noisy data.
- As can be seen in Table 4.2, the fusion of several descriptors does not automatically lead to better results. A finer statistical analysis is needed to exploit the existence of different descriptors.

4.5.2 Limits of the convexity hypothesis

While our approach allows building easily some invariant descriptors of human shape, the application to human motion retrieval using Dynamic Time Warping is a little rough. We can indeed observe in Table 4.5 that the results on the Dyna dataset are outperformed by some of the learned approaches presented above. This can be explained by two factors: first, the

Representation	NN	FT	ST
GDVAE (Aumentado-Armstrong *et al.*, 2019)	18.7	19.6	32.2
Zhou et al. (Zhou *et al.*, 2020a)	50	40.4	57.0
SMPL pose vector	*58.2*	*45.7*	*63.2*
Areas	37.2	24.5	35.8
Breadths	50.7	36.2	50.5
Areas & Breadths	50.7	37.2	51.7

Table 4.5 Results for motion retrieval of translation invariant methods for Dyna dataset.

breadth and areas are symmetric functions, they are not necessarily able to distinguish between the effect of all symmetries that apply to the human pose. Moreover, the convexity hypothesis may be a bit limited. To verify our statement, we test the hypothesis in depth by trying to reconstruct the human pose based on the convex hull. As stated above, the convex hull is characterized by its support function. We test the possibility of reconstructing human poses from support function measurements using state-of-the-art pose priors.

In order to do this, we use VPoser (Pavlakos *et al.*, 2019) combined with the SMPL model. VPoser is a variational autoencoder (a deep learning generative model) trained on the AMASS dataset (Mahmood *et al.*, 2019), which contains more than 11000 human body motions, available as SMPL parameters, and thus is a large dataset of plausible human pose. The network is trained with pose-specific losses to encourage the encoder to penalize impossible human pose generation. The authors of VPoser demonstrate that the network is indeed capable of interpolating between human poses without passing through impossible ones. We search for Vposer-latent codes of human poses using the support function measurements of a shape by minimizing the following function:

$$f_M(z_{\text{poser}}) = \sum_{i=0}^{N} (h(M;u_i) - h(\text{SMPL}(\text{VPoserDecode}(z_{\text{poser}});\beta),u_i)^2 \tag{4.1}$$

where z_{poser} is living in the R^{32} latent space of VPoser and β is a fixed SMPL identity vector. We minimize f_M using gradient descent and show the resulting reconstructions in Figures 4.8 and 4.9. This approach shows that some poses can be almost identically retrieved. However, some failed reconstructions led to "far" human poses (see first and last pose of Figure 4.9).

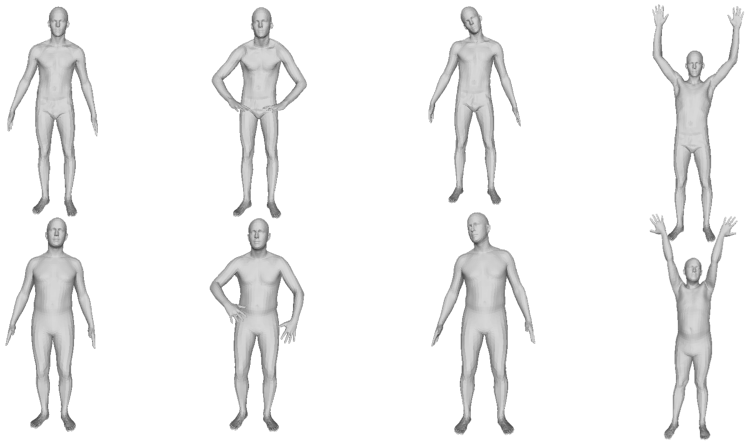


Figure 4.8 Reconstruction of human poses from support function measurements by minimizing Equation (4.1). We see that the poses are not fully recovered.

These results question the convexity hypothesis and may also explain the results presented in Table 4.5.

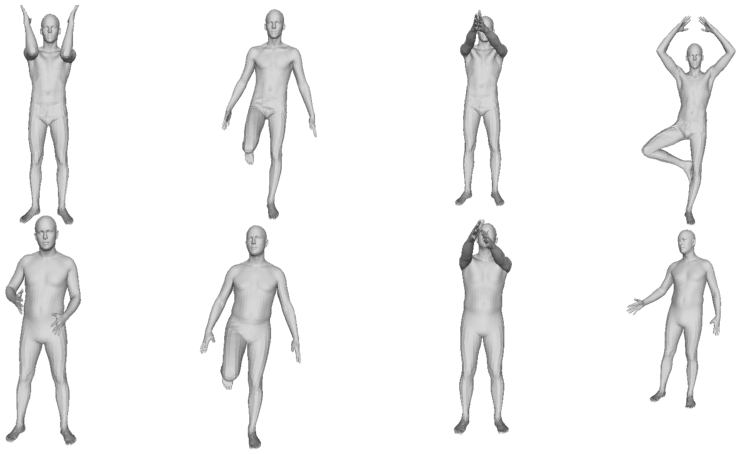


Figure 4.9 Reconstruction from support function measurements by minimizing Equation (4.1). We see that for the second and third poses, the corresponding pose is almost identically retrieved. The optimization failed for the first and last one.

4.6 Conclusion

In this chapter, we defined a novel human descriptor using purely geometric information. Our approach is based on the intuition that a human pose is nearly characterized by its convex hull. Based on this hypothesis, we introduced three sequences of numerical surface descriptors that are invariant under reparametrizations, Euclidean transformations, and similarities. We demonstrated the use of these descriptors by performing pose retrieval and extending their use to human motion retrieval. Our experiments on the FAUST and CVSSP3D synthetic and real datasets demonstrated that our method generally outperforms the state-art-methods for both 3D human pose and motion retrieval including deep learning approaches.

However, we denoted the limits of our approach in the context of noisy shapes. Moreover, the convexity hypothesis seems to be limited in the context of more complex motions. In the next Chapter, we will use another approach coming from the varifold theory to build a human motion descriptor. This will allow us to be robust to noise and to improve our results while keeping the same levels of invariance.

Chapter 5

3D Shape Sequence of Human Comparison and Classification using Current and Varifolds

Contents

5.1 Introduction

The previous Chapter introduced a family of pose descriptors allowing us to compare both human pose and human motion. However, the comparison of motion was stated as a simple comparison of polygonal curves. It does not exploit the underlying characteristics of human motion. In fact, in addition to the non-linearity of the shape space, one expects non-linearity in temporal evolutions, which is causing the difficulties we encountered. Moreover, the experiments on noisy shapes were laborious, with degraded performance. For example, the recent use of the 3D harmonics descriptor (Veinidis *et al.*, 2019), combined with temporal filtering and cautious normalization of human shapes, was efficient on the real dataset (see Table 5.3). Moreover, as stated in the end of the last Chapter, more complex motions like the ones of the Dyna dataset are not well handled by our approach.

In this chapter, we propose to use Gram matrices of motion that we further reduce to Gram-Hankel matrices in order to capture motions' underlying dynamics. As illustrated in Figure 5.1, we first embed the human shape space $\mathscr{H}$ in an infinite dimensional Hilbert space with inner product $\langle\ ,\ \rangle_V$ corresponding to a positive definite kernel inspired by the varifold framework. Using this kernel product we are able to compute the Gram matrix relative to a motion. We then propose to transform each of those matrices, using the Hankel matrices of the motion, into Gram-Hankel matrix of fixed size r. This matrix allows for representing the dynamic of human motion.

In summary, the main contributions of this chapter are:

- We represent 3D human surfaces as varifolds. This representation is equivariant to rotation and invariant to parametrization. This representation allows us to define an inner product between two 3D surfaces represented by varifolds. It is the first use of the space of varifolds in human shape analysis. The framework does not assume that the correspondences between the surfaces are given.

- We represent 4D surfaces by Hankel matrices. This key contribution enables the use of standard computational tools based on the inner product defined between two varifolds. The dynamic information of a sequence of 3D human shape is encapsulated in Hankel matrices and we propose to compare sequences by using the distance between the resulting Gram-Hankel matrices;

- The experiment results show that the proposed approach improves 3D human motion retrieval state-of-the-art and it is robust to noise.

5.1.1 Drawbacks of temporal approaches

As shown in the previous Chapter, a general approach adopted when comparing 3D sequences is the extension of static shape descriptors such as 3D shape distribution, Spin Image or our Shape invariants to include temporal motion information (Huang *et al.*, 2010; Veinidis *et al.*, 2019). While these approaches require the extraction of shape descriptors, our approach does not need a 3D shape feature extraction. It is based on the comparison of surface varifolds within a sequence. In addition, the comparison of 3D sequences requires an alignment of the sequences. The Dynamic Time Warping (DTW) algorithm has been used for several

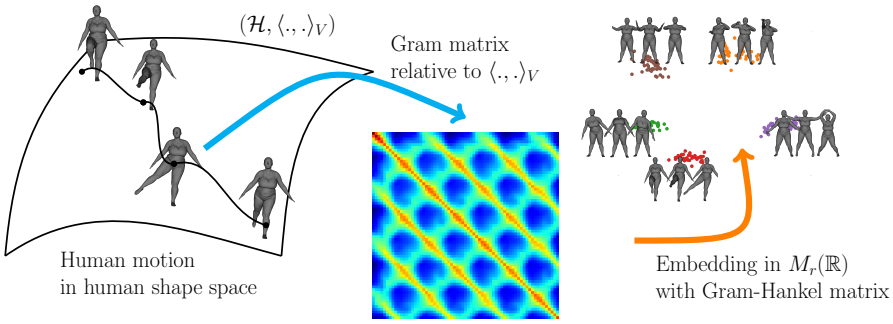


Figure 5.1 Overview of our method. We embed the human shape space $\mathscr{H}$ in an infinite dimensional Hilbert space with an inner product from a positive definite kernel $\langle ., . \rangle_V$ inspired by varifold framework. Using this kernel product we are able to compute the Gram matrix relative to a motion. Each of these Gram matrices is transformed into Gram-Hankel matrix of size r. The Frobenius distance in $M_r(\mathbb{R})$ is used to retrieve similar 3D sequences.

computer vision applications (Ben Amor *et al.*, 2016; Kacem *et al.*, 2018) and alignment of 3D human sequences (Veinidis *et al.*, 2019), as shown before. However, DTW does not define a proper distance (no triangle inequality). In addition, as we have seen, temporal filtering is often required for the alignment of noisy meshes (Slama *et al.*, 2014). Our approach enables the comparison of sequences of different temporal duration, does not need any alignment of sequences, and is robust to noisy data. We model a sequence of 3D mesh as a dynamical system. The parameters of the dynamical system are embedded in our Hankel matrix-based representation. Hankel matrices have already been adopted successfully for skeleton action recognition in (Zhang *et al.*, 2016). As we do not have finite-dimensional features to build such matrices numerically, we define the novel Gram-Hankel matrix, based on the kernel product defined from the surface varifold. This matrix is able to model the temporal dynamics of the 3D meshes.

5.2 Varifolds for human shapes

To work with varifolds for human shapes, we follow the work of (Charon & Trouvé, 2013; Kaltenmark *et al.*, 2017). They designed a fidelity metric based on the varifold representation and Reproducing Kernel Hilbert Spaces. This fidelity metric is proposed for 2D artificial contour retrieval, and used in 3D diffeomorphic registration in the Large deformation diffeomorphic metric mapping (LDDMM) framework. To our knowledge, the present work is the first use of such representation for the analysis of human shape. It is also the first use of the space of varifolds purely for itself, as an efficient way to perform direct computations on shapes.

5.2.1 Choice of reproducing kernel

The comparison of measures themselves is, however, a though problem, where one would need to define discrepancy from optimal transport or define a large set of well-defined functions on which to apply the measure. A more natural way is to look in depth at the theory of Reproducing Kernel Hilbert Spaces (RKHS). The underlying idea is to embed any varifold of

a surface M with a representant Φ_M living in a (infinite dimensional) Hilbert space, which has a well-defined inner product, allowing to define distances between human shapes;

A varifold μ can be converted into a function Φ_μ on $\mathbb{R}^3 \times \mathbb{S}^2$ using a *reproducing kernel* that comes from the product of two positive definite kernels: $k_{pos} : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}$, and $k_{or} : \mathbb{S}^2 \times \mathbb{S}^2 \to \mathbb{R}$. We just define $\Phi_\mu(x,v) = \int_{\mathbb{R}^3 \times \mathbb{S}^2} k_{pos}(y,x)k_{or}(w,v)d\mu(y,w)$. For a triangulated surface M, we get $\Phi_M(x,v) = \sum_{i=1}^m a_i k_{pos}(c_i,x)k_{or}(n_i,v)$.

One obtains a Hilbert product between any two varifolds μ, ν as follows :

$$\langle \mu, \nu \rangle_V = \left\langle \Phi_\mu, \Phi_\nu \right\rangle_V = \int \Phi_\mu d\nu = \int \Phi_\nu d\mu,$$

so that

$$\langle \mu, \nu \rangle_V = \iint k_{pos}(x,y)k_{or}(v,w)d\mu(x,v)d\nu(y,w).$$

We deduce the explicit expression for that product between two triangulated 3D shapes M and N :

$$\langle M, N \rangle_V = \langle \Phi_M, \Phi_N \rangle_V = \sum_{i=1}^{m}\sum_{j=1}^{n} a_i^M a_j^N k_{pos}(c_i^M, c_j^N)k_{or}(n_i^M, n_j^N) \tag{5.1}$$

Where m, n is the number of faces of M and N. The continuous version of this product presented in (Charon & Trouvé, 2013) is parametrization invariant.

A nice property of such a product is that it can be made equivariant to rigid transformation by carefully choosing the kernels. First, we define how to apply such a deformation on a varifold. Given a rotation $R \in SO(3)$ of $\mathbb{R}^3$ and a vector $T \in \mathbb{R}^3$, the rigid transformation $\psi : x \mapsto Rx+T$ yields the push-forward transformation $\mu \mapsto \psi_\# \mu$ through $\int f d\psi_\# \mu = \int \phi(Rx+T, Rv)d\mu$ on the space of varifolds. For a triangulated surface M, $\psi_\# M$ is just $\psi(M)$, the surface obtained by applying the rigid motion ψ to the surface M itself. We have the following important result :

Theorem 5.2.1. *If we define the positive definite kernels as follows:*

$$\begin{aligned} k_{pos}(x,y) &= \rho(||x-y||), \quad x,y \in \mathbb{R}^3, \\ k_{or}(v,w) &= \zeta(v.w), \quad v,w \in \mathbb{S}^2, \end{aligned}$$

then for any two varifolds μ, ν, and any rigid motion ψ on $\mathbb{R}^3$, we have

$$\langle \psi_\# \mu, \psi_\# \nu \rangle_V = \langle \mu, \nu \rangle_V.$$

This result means that given a rigid motion ψ, $\langle \phi(M), \phi(N) \rangle_V = \langle M, N \rangle_V$.

The kernel k_{pos} is usually chosen as the Gaussian kernel $k_{pos} = e^{-\frac{||x-y||^2}{\sigma^2}}$, with the scale parameter σ needed to be tuned for each application.

Kaltenmark *et al.* (Kaltenmark *et al.*, 2017) proposed several function for the spherical kernel. In this chapter we retained the following functions:

- *currents*, with $\zeta(u) = u$
- *oriented varifolds*, with $\zeta(u) = e^{2u/\sigma_o^2}$
- we finally propose *absolute varifolds*, with $\zeta(u) = |u|$

5.2.2 Pertinence of varifolds in the human shapes context

For such kernels, two surface varifolds M, N with "similar" support (for example, if M is a reparametrization of N, or if they represent two human shapes with the same pose but different body types) will have relatively small distance in the space of varifolds, so that $\langle M,N\rangle_V^2 \simeq \langle M,M\rangle_V \langle N,N\rangle_V$, that is, they are almost co-linear. On the other hand, surface varifolds with very distant support will be almost orthogonal ($\langle M,N\rangle_V \simeq 0$) because of the Gaussian term in k_{pos}. Obviously, shapes that have some parts that almost overlap while others are far away will be in-between. Combined with its rotational equivariance, this leads us to believe that the kernel product can be used to differentiate between poses and motions independently of body types. However, as we can see in Table 5.1, the distance does not differentiate well human poses and obtain poor results in pose retrieval.

Distance	NN	FT	ST
Currents	12	12.3	18.7
Absolute Varifolds	11	12.9	19.6
Oriented Varifolds & Breadths	11	12.0	18.9

Table 5.1 Retrieval results of the varifold distance on the FAUST dataset, using the varifold distances (using $\sigma = 0.1$, similar results with other values). The results are poor.

Connections with the previous chapter. We can see a few links in the approach of the previous Chapter and this one. First, varifolds are measures of functions over $\mathbb{R}^3 \times \mathbb{S}^2$ and the weighted area function exploits the EGI of the surface, which can be seen as a measure of functions over $\mathbb{S}^2$. Moreover, using varifolds, we transform them into functions $\mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$. Moreover, our extraction of the human pose feature vector was a sampling of a function $\mathbb{S}^2 \to \mathbb{R}$. Thus varifolds contain more information than the previous approach. However, under the convexity hypothesis, the pose feature vector showed to be more adapted than the varifold distance for pose description as we see in table 5.1. While the kernel metric of varifold seems inadapted in this context, we can take advantage of the high dimensional nature of the varifolds RKHS to define a convenient motion descriptor.

5.3 Comparing 3D Human Sequences

We need a way to compare sequences of $3D$ shapes $M_1, ..., M_T$, with T possibly differing between sequences. For this, we use the kernel product $\langle\ ,\ \rangle_V$ as a similarity metric. Thanks to the reproducing property of positive definite kernels (Aronszajn, 1950), it defines a reproducing kernel Hilbert space $\mathscr{H}$ (RKHS) which is an (infinite dimensional) Euclidean space endowed with an inner product corresponding to the kernel product, as described in the previous section. Any shape M has a corresponding representative $\Phi_M : \mathbb{R}^3 \times \mathbb{S}^2 \to \mathbb{R}$ in this space, such that $\langle \Phi_M, \Phi_N\rangle_V = \langle M,N\rangle_V$ (Figure 5.2).

Modeling dynamics of temporal sequences. Thanks to the varifolds representation and the kernel product $\langle .,. \rangle$, the temporal sequence $M_1, ..., M_t$ corresponding to a motion in the human shape space can be seen as a temporal sequence $\Phi_{M_1}, ..., \Phi_{M_T}$ in the RKHS $\mathscr{H}$. Plus, since the varifold kernel is equivariant to rigid transformations, the product of two shapes within

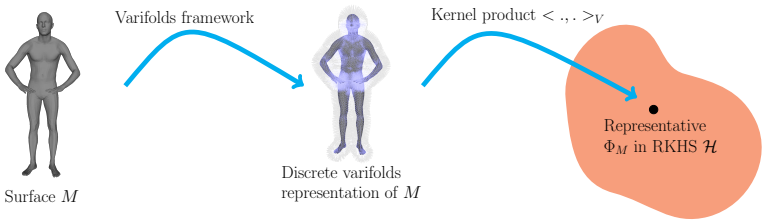


Figure 5.2 An overview of varifolds framework. First, a mesh M is transformed into its corresponding varifold representation. Then, the kernel product defined in Equation (5.1), transforms it into a representant Φ_M living in the Hilbert space $\mathscr{H}$ of this varifold.

a sequence is invariant to any rigid transformation *applied to the full motion*. The Gram matrix $J_{ij} = \langle \Phi_{M_i}, \Phi_{M_j} \rangle$, which is a rigid transformation invariant matrix, would be a natural representant of the motion. However, its size varies with the length T of the sequence. Inspired by Auto-Regressive (AR) models of complexity k defined by $\Phi_{M_t} = \sum_{i=1}^{k} \alpha_i \Phi_{M_{t-i}}$, several representations (Slama *et al.*, 2015; Turaga *et al.*, 2008) have been proposed for dynamical systems modeling. Hankel matrices (Li *et al.*, 2012) are one of the possible representations. The Hankel matrix of size r, s corresponding to our time series $\Phi_{M_1}, ..., \Phi_{M_T}$ is defined as:

$$\mathbf{H}_t^{r,s} = \begin{pmatrix} \Phi_{M_1} & \Phi_{M_2} & \Phi_{M_3} & ... & \Phi_{M_s} \\ \Phi_{M_2} & \Phi_{M_3} & \Phi_{M_4} & ... & \Phi_{M_{s+1}} \\ ... & ... & ... & ... & ... \\ \Phi_{M_r} & \Phi_{M_{r+1}} & \Phi_{M_{r+2}} & ... & \Phi_{M_{r+s}} \end{pmatrix} \tag{5.2}$$

The rank of such matrix is usually, under certain conditions, the complexity k of the dynamical system of the sequence. The comparison of two time series therefore become a comparison of high dimensional matrices. Such an approach has been applied successfully to skeletal and video action analysis, with various proposed ways for comparing Hankel matrices (Zhang *et al.*, 2016; Lo Presti & La Cascia, 2015; Li *et al.*, 2012).

It is not straightforward to use those matrices since our shape representatives live in infinite dimensional space. A first idea would be to think about the Nystrom reduction method (Williams & Seeger, 2001) to build an explicit finite dimensional representation for Φ_M, but this would involve intensive computations on the full Gram matrix of the sequence pairwise inner products. Another possibility is to think about the Gram matrix $\mathbf{H}\mathbf{H}^T$ derived from the Hankel matrix $\mathbf{H}$ (Zhang *et al.*, 2016; Li *et al.*, 2012). We cannot directly derive the same kind of matrices since our representatives live in an infinite dimensional space. The Gram matrix of the motion, J, however, preserves the linear relationships of the AR model. We therefore derive the following matrix:

Definition 5.3.1. *The Gram-Hankel matrix of size r, $G \in M_r(\mathbb{R})$ of the sequence $\Phi_{M_1}, ..., \Phi_{M_T}$ is defined as:*

$$\mathbf{G}_{ij} = \sum_{k=1}^{T-r} \langle \Phi_{M_{i+k}}, \Phi_{M_{j+k}} \rangle = \sum_{k=1}^{T-r} \langle M_{i+k}, M_{j+k} \rangle_V \tag{5.3}$$

We normalize $\mathbf{G}$ relatively to the Frobenius norm, following recommended practices (Zhang *et al.*, 2016). This matrix is the sum of the diagonal blocks B_l^r of size r of the Gram matrix

of the sequence pairwise inner products. A possible way of interpreting what encodes a single block B_l^r of size r when $r \geq k$ is to follow the idea of (Kacem *et al.*, 2018) the polar decomposition of the coordinate matrix of $\Phi_{M_l}, ...\Phi_{M_{r+l}}$. This coordinate matrix exists in the space $span(\Phi_{M_0}, ...\Phi_{M_k})$ under to the AR model hypothesis (any Φ_{M_j} is a linear combination of the first k Φ_{M_i}), and can be factorized into the product $U_l R_l$, where U_l is an orthonormal $r \times k$ matrix, and R_l an SPD matrix of size k. The matrix R_l is the covariance (multiplied by r^2) of $\Phi_{M_l}, ...\Phi_{M_{r+l}}$ in $span(\Phi_{M_0}, ...\Phi_{M_k})$, and it encodes in some way its shape in this space. An illustration of such encoding is given in Figure 5.3. For three motions from CVSSP3D dataset, we compute the varifold distances Equation (5.1) between all samples of the motion. We then used Multidimensional Scaling (MDS) (Cox & Cox, 2008) to visualize them in a 2D space. We display the ellipse associated to the covariance of each motion. We see that the one associated to jump in place (blue) motion is distinguishable from the ones associated to walk motions (red and green).

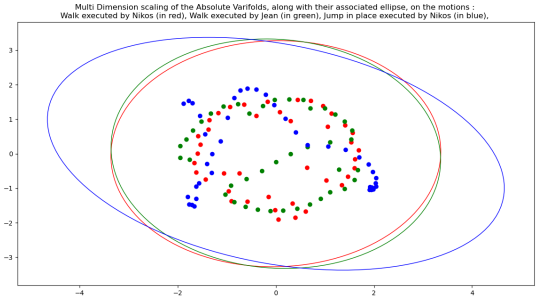


Figure 5.3 MDS illustration of three motions of CVSSP3D dataset, along with ellipse associated to their covariance.

The Gram matrix block B_l^r is written as $B_l^r = U_l R_l^2 U_l^T$ and contains such information. Searching for the complexity k of the AR model would be sensitive to errors, and computing the associated R_l for comparisons with an SPD metric would be time consuming in our case. We thus preferred the rather simpler Gram-Hankel matrix, that cancels possible noise in single blocks when summing them. Finally, using the Frobenius distance $d(\mathbf{G_i}, \mathbf{G_j}) = ||\mathbf{G_i} - \mathbf{G_j}||_F$ where G_i and G_j are two Gram-Hankel matrices, to compare two motions lead us to rather good results. The blocks of size r are expressive enough when $r \geq k$, and taking their sum will ensure us to cancel possible noise added to a single block. The degenerate nature of G, does not allow for efficient use of SPD metrics such as the Log-Euclidean Riemannian Metric (LERM) on the G_i. With this approach, the comparison of two human motions is formulated as the comparison of two symmetric positive semi-definite matrices.

Proposition 5.3.2. *The Gram-Hankel matrix G associated to a motion $M_1, ..., M_T$ defined by Equation (5.3) has the following properties:*

1. *It is invariant to parameterization (property of the kernel product).*

2. *It is invariant to rigid transformation applied to a motion.*

Normalizations. As the definition of the kernel shows the method is not invariant to scale, we normalize the inner products as following: $\left\langle \frac{M_i}{||M_i||_V}, \frac{M_j}{||M_j||_V} \right\rangle_V$.

While our method is translation invariant, the use of the Gaussian kernel implies that the product will be near 0 when the human shapes are at long range. To avoid this, we translate the surface M with triangles $T_1, ..., T_m$ by its centroid $c_M = \frac{\sum_{i=1}^{m} a_i c_i}{\sum_{i=1}^{m} a_i}$, where c_i and a_i correspond to the center and area of triangle T_i. We apply $M \mapsto M - c_M$ before computing the products.

5.4 Experiments

Computing varifold kernel products can often be time consuming, due to the quadratic cost in memory and time in terms of vertex number for computing $\langle M, N \rangle_V$. However, the recent library Keops (Charlier *et al.*, 2021), designed specifically for kernel operations proposes efficient implementations with no memory overflow, reducing time computation by two orders of magnitudes. We used those implementations with the Pytorch backend on a computer setup with Intel(R) Xeon(R) Bronze 3204 CPU @ 1.90GHz, and a Nvidia Quadro RTX 4000 8GB GPU.

5.4.1 Evaluation setup

In order to measure the performance in motion retrieval, we use the classical performance metrics used in retrieval: Nearest neighbor (NN), First-tier (FT) and Second-tier (ST) criteria (see previous chapter for definition). For each experiment, we take r values ranging from 1 to T_{min} where T_{min} is the minimal sequence length in the dataset. We also take 10 σ values for the Gaussian kernel ranging from 0.001 to 10 in log scale. The score displayed is the best score among all r and σ values. For oriented varifolds, the σ_o of the gamma function is fixed to 0.5 as in (Kaltenmark *et al.*, 2017).

5.4.2 Datasets

The datasets chosen are the following :

- The CVSSP3D synthetic dataset in the same configuration as last chapter
- The CVSSP3D real dataset in the same configuration as last chapter
- The 144 motions of Dyna dataset are use for motion retrieval.

5.4.3 Comparison with state-of-the-art.

We proceed to an extensive comparison to state-of-the-art descriptor of human body. The polygonal curves of those descriptors are combine with dynamic time warping distances, and filtered with a temporal filtering approach for the real dataset. The descriptors of the comparison are the following :

1. Spin Images and Shape Distribution are part of the comparison

2. The 3D harmonics descriptor (Papadakis *et al.*, 2008)(Veinidis *et al.*, 2019) is a descriptor based on point cloud repartition in space. A 3D shape is first normalized with two variations of PCA. Then, a spherical histogram with different rays is built. The final descriptor is decomposed along spherical harmonics of the obtained with a specific re-weighting for better results. Temporal filtering is proposed in order to deal with the real dataset.

3. Breadths spectrum Q-breadths and Q-shape invariant from the previous chapters.

4. We re-use the pose latent vector from Geometrically Disentangled Variational AutoEncoder (GDVAE) of (Aumentado-Armstrong *et al.*, 2019).

5. We also re-use the posent latent vector of Zhou *et al.* (Zhou *et al.*, 2020a) Neural3DMM (Bouritsas *et al.*, 2019) disentangled network.

6. Cosmo *et al.* (Cosmo *et al.*, 2020) propose a human pose latent vector in a similar approach as GDVAE, called Latent Interpolation with Metric Priors (LIMP). They use the same type of autoencoder as GDVAE but change the disentanglement constraints with metric prior constraints: a change in extrinsic latent space should only induce change on extrinsic distances of the meshes, while a change in intrinsic latent space should only induce change on intrinsic distances of the meshes. They use Euclidean and geodesic pairwise matrices in their losses to model this constraint, which needs a constant parameterization in the training set. We use the network pretrained on the FAUST dataset (Loper *et al.*, 2015). They do not make any specific training for Euclidean invariance. In order to do motion retrieval, we applied the meshes as input of their available trained network and gathered their extrinsic latent vectors (belonging to $\mathbb{R}^{64}$), and used them in the human sequence retrieval pipeline.

7. We also compare to the SMPL registrations available in the Dyna dataset. The SMPL parameters were augmented with dynamic soft tissue deformation relative to each motion (called DMPL) and use to transform the original Dyna dataset to the DFAUST dataset, with better correspondance with the scan. They use for this goal much more information such as texture information from body videos, and the shape vector is retrieved using gender information. We prefer comparing on Dyna dataset rather than DFAUST dataset, allowing us to compare faithfully to the SMPL body pose descriptor. In order to build the pose vectors, a costly fitting method is used along each sequence (accounting in minutes for a single shape). The pose vectors for 129 motions of Dyna where the fitting was successful, we added the SMPL Pose vector retrieved using available code `https://github.com/vchoutas/smplx/` for the remaining 5 motions.

5.4.4 Motion Retrieval on CVSSP3D artificial dataset

For the artificial dataset, the optimal σ were fixed to 0.17 for current, 0.17 for absolute varifolds, and 0.02 for oriented varifolds. The optimal r were 97, 92 and 90 for current, absolute and oriented varifolds respectively. The maximum computation time of Gram-Hankel matrix is 0.89s.

Representation	Diff$^+$ inv.	$SO(3)$ inv.	NN	FT	ST
Shape Dist.	✓	✓	92.1	88.9	97.2
Spin Images	✓	✓	100	87.1	94.1
3D harmonics	≈	≈	100	98.3	99.9
Breadths spectrum	✓	✓	100	99.8	100
Shape invariant	✓	✓	82.1	56.8	68.5
GDVAE	✓	✓	100	97.6	98.8
Zhou *et al.*	✗	✗	100	99.6	99.6
LIMP	✓	✗	100	*99.98*	*99.98*
Current	✓	✓	100	100	100
Absolute varifolds	✓	✓	100	100	100
Oriented varifolds	✓	✓	100	100	100

Table 5.2 CVSSP3D artificial dataset results for motion retrieval. Results of Shape Distributions, Spin Images and 3D harmonics are taken from (Veinidis *et al.*, 2019).

We observe the results on the CVSSP3D artificial dataset in Table 5.2. Only our approach is able to get 100% in all performance metrics. We also observe that only our approach able to outperform the LIMP learned approach.

5.4.5 3D Human Motion retrieval on CVSSP3D real dataset

For the real dataset, the optimal σ were fixed to 0.06 for current, 0.17 for absolute varifolds and oriented varifolds. The optimal r were 48, 43 and 46 for current, absolute and oriented varifolds respectively. The maximum computation time of Gram-Hankel matrix is 6m30s (the meshes of the dataset are much larger). We emphasize that as opposed to any other approaches for this dataset, we do not apply any kind of adaptation for the noise present in the meshes. We observe the results on the CVSSP3D real dataset in Table 5.3. Absolute varifolds approach

Representation	Diff$^+$ inv.	$SO(3)$ inv.	NN	FT	ST
Shape Dist.	✓	✓	77.5	51.6	65.5
Spin Images	✓	✓	66.3	43.2	59.5
3D harmonics	≈	≈	92.5	**72.7**	**86.1**
Q-Breadths spectrum	≈	✓	80.0	44.8	59.5
Q-shape invariant	≈	✓	82.5	51.3	68.8
GDVAE	✓	✓	38.7	31.6	51.6
Current	✓	✓	92.5	66.0	78.5
Absolute varifolds	✓	✓	**95.0**	66.6	80.7
Oriented varifolds	✓	✓	93.8	65.4	78.2

Table 5.3 CVSSP3D artificial dataset results for motion retrieval. Results of Shape Distributions, Spin Images and 3D harmonics are taken from (Veinidis *et al.*, 2019).

outperforms by 2.5% the 3D descriptor in terms of NN metric, while being less good for FT and ST. In terms of fully invariant methods, we outperform by 10% the proposed approaches. The absolute varifolds methods is the best of our approach, but we do not observe significant

sensitivity between different varifolds. We finally observe that the point cloud descriptors of GDVAE has the lowest performance.

5.4.6 3D Human Motion Retrieval on Dyna dataset

For the Dyna dataset, the optimal σ were fixed to 0.02 for current 0.06 for absolute varifolds, and 0.16 for oriented varifolds. The optimal r were 27, 31 and 72 for current, absolute and oriented varifolds respectively. The maximum computation time of Gram-Hankel matrix is 2m30s.

Representation	Diff$^+$ inv.	$SO(3)$ inv.	NN	FT	ST
Areas	✓	✗	37.2	24.5	35.8
Breadths	✓	✗	50.7	36.2	50.5
Areas & Breadths	✓	✗	50.7	37.2	51.7
GDVAE	✓	✓	18.7	19.6	32.2
Zhou *et al.*	✗	✗	50.0	40.4	57.0
LIMP	✓	✗	29.1	20.7	33.9
SMPL pose vector	≈	✓	*58.2*	***45.7***	***63.2***
Current	✓	✓	59.0	34.1	50.4
Absolute varifolds	✓	✓	**60.4**	40.0	55.9
Oriented varifolds	✓	✓	**60.4**	40.8	55.9

Table 5.4 Dyna dataset results for motion retrieval.

As shown in Table 5.4 the oriented and absolute varifolds is the best by 2 % in terms of NN metric compare to SMPL, and by more than 10 % to other approaches, including the parameterization dependant approach of (Zhou *et al.*, 2020a). The FT and ST performance are however less good than SMPL. This can be explained by its human specific design, along with the costly fitting method, that use additional information (gender, texture videos). Finally, we observe that point cloud neural networks are not suitable for high set of complex motions.

5.4.7 Qualitative results

Confusion matrices

We display in Figure 5.4 the Nearest Neighbor score confusion matrices for both SMPL and Oriented Varifolds. We observe that on Dyna, the difficult cases were *jiggle on toes, shake*

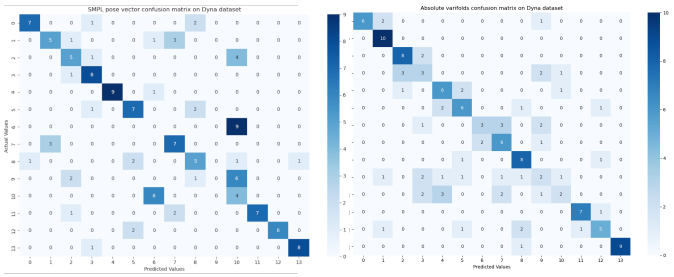


Figure 5.4 Confusion matrix of SMPL (left) and Oriented Varifolds (right) on the Dyna dataset.

arms, shake hips and jumping jacks, corresponding to l3, l6, l9 and l10 in confusion matrix. Our approach is able to classify better these motions than SMPL. In addition, SMPL was not able to retrieve as a Nearest Neighbor, a similar motion to shake arms or shake hips corresponding to l6 and l9. This Figure shows also that our approach retrieves perfectly the knees motion corresponding to l1.

Example queries

To augment the qualitative comparisons, we propose two queries on Dyna and CVSSP3D real datasets. The results on Dyna, displayed in Figure 5.5, show the good quality of our approach for the chosen query. We observe that Areas & Breadths are gathering too much symmetric information, and that SMPL retrieves some motion executed by the same individual. Our approach does not display such problems.

Moreover, the results on CVSSP3D, displayed in Figure 5.6 shows an improvement compared to Q-breadths spectrum, a fact that confirms the results of Table 5.3.

5.5 Discussion

5.5.1 Comparison of SPD metrics for Gram-Hankel matrices

As a matter of fact, the Gram-Hankel matrices are positive semidefinite matrices, and several metrics have been proposed to compare positive definite matrices. We display in Table 5.5 the results of on of them, the Log Euclidean Riemannian Metric (LERM) compared to the Frobenius distance.

$$d_{LERM}(G_1, G_2) = ||\log(G_1) - \log(G_2)||_F,$$

where $\log(G) = P^T \log(\lambda) P$, where $G = P^T \lambda P$ is the eigen decomposition of the symmetric matrix G. We observe that on all datasets, the performance is lower than using the Frobenius

Representation	Gram-Hankel distance	Artificial dataset			Real dataset			Dyna dataset		
		NN	FT	ST	NN	FT	ST	NN	FT	ST
Current	Frobenius	100	100	100	92.5	66.0	78.5	59.0	34.1	50.4
	LERM	100	100	100	78.8	55.0	76.6	55.2	35.9	51.4
Absolute varifolds	Frobenius	100	100	100	95.0	66.6	80.7	60.4	40.0	55.9
	LERM	100	100	100	80.0	54.6	73.4	57.5	36.0	50.8
Oriented varifolds	Frobenius	100	100	100	93.8	65.4	78.2	60.4	40.8	55.9
	LERM	100	100	100	86.3	50.0	66.4	57.5	37.0	51.3

Table 5.5 Motion retrieval results for our approach with Log Euclidean Riemannian Metric (LERM). The results are displayed for CVSSP3D artificial and real datasets, and Dyna datasets

metric. This results confirm our choice of using Frobenius than LERM metric.

5.5.2 Effect of the parameter choice for oriented varifolds on Dyna dataset.

Effect of the sigma parameter. The performance relative to the σ parameter is displayed on the right of Figure 5.7 for oriented varifolds on Dyna dataset. We observe first that the choice of σ has a significant impact on performance for NN and in the same time that the optimal σ for the NN is not the same as one for FT and ST, for a loss of around 2% in those metrics,

(a) Query using Areas & Breadths

(b) Query using SMPL

(c) Query using oriented varifolds.

Figure 5.5 First tier of the query "50020, running on spot" for Areas & Breadths, SMPL (Loper *et al.*, 2015) and oriented varifolds on the Dyna dataset. The query is in yellow and the results are sorted by closeness to the query using a given approach.

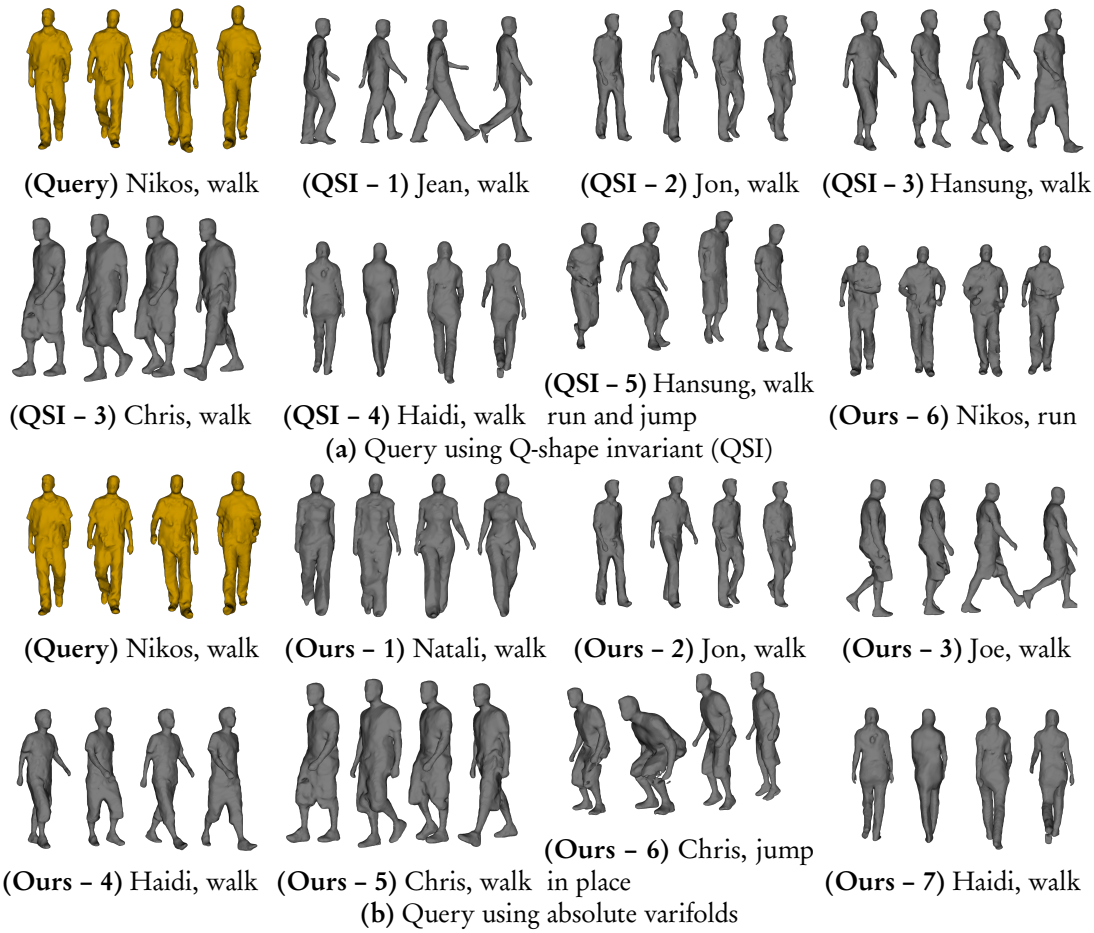


Figure 5.6 First tier of the query "Nikos, Walk" for Q-shape invariant and Absolute Varifolds on the CVSSP3D real dataset. The query is in yellow and the results are sorted by closeness to the query using a given approach. The first query is directly taken from the previous chapter.

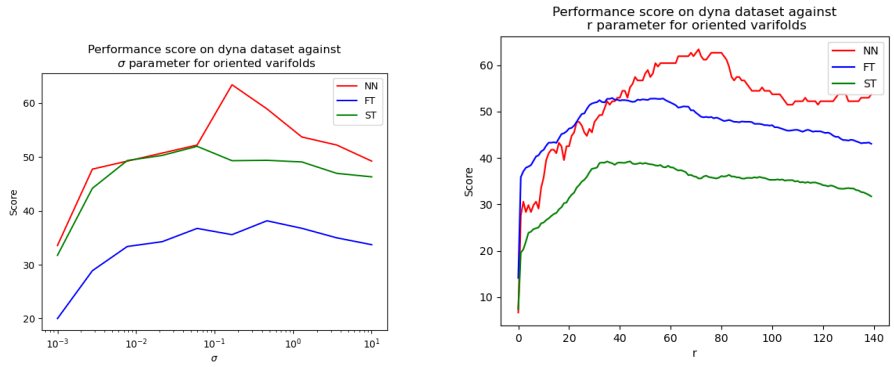


Figure 5.7 NN, FT, ST metric relatively to the σ parameters (left) and to the r parameter (right) on Dyna dataset for oriented varifolds.

which is less significant than the NN gain.

Effect of the choice of r. The performance relative to the r parameter is displayed on the left of Figure 5.7 for oriented varifolds on Dyna dataset. We observe first that the choice of r has a significant impact on performance and in the same time that the optimal r for the NN is not the same as one for FT and ST, for a loss of around 5% in those metrics.

Centroid	Inner	NN	FT	ST
✗	✗	51.5	34.6	53.4
✗	✓	52.2	33.4	50.5
✓	✗	59.7	40.7	55.8
✓	✓	**60.4**	**40.8**	**55.9**

Table 5.6 Retrieval performance of the normalizations on Dyna dataset, for oriented varifolds. Both are useful.

Effect of normalizations. We present in Table 5.6 the performances of oriented varifolds with the 2 normalization techniques presented here. The centroid normalization is essential to the good performance of our approach. In the mean time, the inner product normalization always implies significant boost for NN metric, but can induce a (non-significant) loss in ST and FT metrics. We provide in, the performance relative to the parameters σ and r, for oriented varifolds on Dyna dataset. We observe that the choice of those parameters is crucial. We also display the performances of oriented varifolds with the 2 normalizations techniques, showing that they both help to obtain the best results.

5.6 Limitations.

In the current setting, we note that our approach presents two main limitations, that open interesting future directions:

1. To measure distance between matrices, we have used Euclidean distance, which does not exploit the geometry of the symmetric positive semi-definite matrices manifold. Metrics on the SPD manifold (different from semi-definite ones) seem to be however unadapted as stated in the discussion section.

2. There is no theoretical limitation to applying this framework to the comparison of other 3D shape sequences (eg. 3D facial expressions, or 3D cortical surface evolutions) other than between body shapes. However, in practice, one should redefine the hyperparameters (r, σ) of the Kernel (Theorem 1).

5.7 Conclusion

We presented a novel framework to perform comparison of 3D human shape sequences. We propose a new representation of 3D human shape, equivariant to rotation and invariant to parameterization using the varifolds framework. We propose also a new way to represent human motion by embedding the 3D shape sequences in infinite dimensional space using a

kernel positive definite product from the varifold framework. We compared our method to the combination of dynamic time warping and static human pose descriptors. Our experiments on 3 datasets showed that our approach gives competitive or better than state-of-the-art results for 3D human motion retrieval, showing better generalization ability than popular deep learning approaches.

However, while this chapter and the previous one proposed efficient approaches for 3D shape sequence retrieval, they are not suited for the problem of human deformations since we leave the shape space in each of them. We will focus on this problem in the next two chapters.

Chapter 6

A Riemannian Framework for Analysis of Human Body Surface

Contents

6.1 Introduction

This Chapter and the next will focus on Human Body deformation, with the aim to open possibilities of applications in vision, graphics, virtual reality, product design, and avatar creation. We remind that human bodies vary significantly across two important properties: shape (or subject identity) and body pose (articulation of the body in space). These variations are what make human body shape analysis a challenging problem. In this Chapter, we seek a framework for human shape analysis that provides:

1. a shape metric to quantify shape and pose differences
2. a full pipeline for generating deformations and shape interpolation;
3. a shape summary, a compact representation of human shapes in terms of the center (mean of human shapes)

We solve this problem by using a Riemannian approach. The core of our approach is the mapping of the human body surface to the space of metrics and normals. We equip this space with a family of Riemannian metrics, called Ebin (or DeWitt) metrics. This family of metrics is invariant under rigid motions and reparametrizations; hence it induces a metric on the shape space. Using the alignment of human bodies with a given template, we show that this family of metrics allows us to distinguish the changes in shape and pose. The proposed framework has several advantages. First, we define a family of metrics with desired invariance properties for the comparison of human shape. Second, we present an efficient framework to compute geodesic paths between human shapes given the chosen metric. Third, this framework provides some basic tools for statistical shape analysis of human body surfaces. Finally, we demonstrate the utility of the proposed framework in pose and shape retrieval of the human body.

6.1.1 Drawbacks of elastic shape analysis

We remind that elastic shape analysis (Kurtek *et al.*, 2012; Tumpach *et al.*, 2016) consists of seeing the shape space defined in chapter 3, as an (almost) infinite-dimensional differentiable manifold. We can then define and use Riemannian metrics on this manifold, seen as a submetric from the surface space, to measure the distance between two given shapes as well as to interpolate between them by computing a geodesic that joins them. However, in classic elastic shape analysis, the model shape is chosen to be the sphere $\mathbb{S}^2$, not a human template, and thus assumes that the surfaces are given by analytic (spherical) representations. Moreover, solving the geodesic problem over paths and space of reparameterizations and rotations is a tedious task, for which the computational costs of this approach are high. The *square root normal fields* or SRNF approach ((Jermyn *et al.*, 2012)) allows to solve efficiently those two problems but is neither injective nor surjective and there exist different shapes having the same SRNF (Bauer *et al.*, 2022). In addition, as observed by Su et al. (Su *et al.*, 2020b), the resulting distance can be viewed as an extrinsic distance, losing essential intrinsic information. Moreover, we will see that the deformations generated by the SRNF metric are poor compared to richer metrics.

In Chapter 3 we define our model shape, from which we define surfaces, as a human template. In this section, we ditch the need for a spherical parametrization by working on registered shapes (which, as we will see, is equivalent to working on the shape space) to a

template mesh. Thus we focus on a framework for the generation of plausible deformations and statistical analysis and leave the parameterization problem for the next Chapter. To this aim, we will take advantage of the recent developments of static and dynamic human datasets such as FAUST and D-FAUST to generate deformations of human shapes.

6.1.2 Main contributions

In this Chapter, we present a comprehensive Riemannian framework for analyzing human bodies, in the process of dealing with the change in shape and pose. We use the Dewitt family of Riemannian, which is a subsect of the broader Sobolev family of metrics, and where we can see surfaces as maps to normals and Riemannian metrics (first fundamental forms). To our best knowledge, this is the first demonstration of the use of this metric in human body shape analysis. We will show also for the first time, that this family of metrics takes into account the intrinsic and extrinsic geometry of human bodies, i.e their shape and pose. Additionally, we present an efficient framework to compute geodesics between two given human body surfaces under the chosen metric. We provide some basic tools for statistical shape analysis of human body surfaces. These tools help us to compute the average human body. To evaluate our approach, we conduct extensive experiments on the FAUST dataset. The experimental results show that the proposed family of Riemannian metrics classifies correctly the shapes and the poses. The experimental results show also that our proposed framework provides better geodesics than the state-of-the-art Riemannian framework.

6.2 Shape Space as the space of aligned Human Bodies

In chapter 3, we defined our shape space as the quotient space of surfaces by the group G of preserving transformations, which is in this chapter $G = \mathbb{R}^+ \times \mathbb{R}^3 \times SO(3) \times \Gamma$. We treat the case of $\mathrm{Diff}^+(\mathcal{T})$ separately. In this chapter, each human body surface is aligned to a given template $\mathcal{T}$. This means that for any equivalence class $[f] \in \mathcal{H}/G$ a preferred correspondence with the template is chosen. This alignment is meaningful both geometrically (for instance the fingertips of the template correspond to the fingertips of the other human bodies, as illustrated in Figure 6.1), which means that geodesics on this surface subspace are horizontal, and anatomically, which means that template parameterization (SMPL template) emphasizes high curvatures points such as the fingertips. We work thus on the set of aligned human bodies, denoted by $\mathcal{S}_0$.

Mathematically the choice of a preferred alignment with the template is called a section $\mathcal{S}_0$ of the fiber bundle $\Pi : \mathcal{H} \to \mathcal{H}/G$. A section of Π is a (smooth) map assigning to each equivalence class $[f_0] \in \mathcal{H}/G$ a representative $f_0 \in \mathcal{H}$ in this class, i.e. such that $\Pi(f_0) = [f_0]$. This notion is illustrated in Figure 6.2. The section we are using, i.e. the correspondence with the template, is smooth as explained above. An illustration of the section is displayed in Figure 6.2.

In this Chapter, we pull back the Riemannian metrics that are defined on shape space (see in section 4) on the preferred section $\mathcal{S}_0$ given by the correspondence with the template.

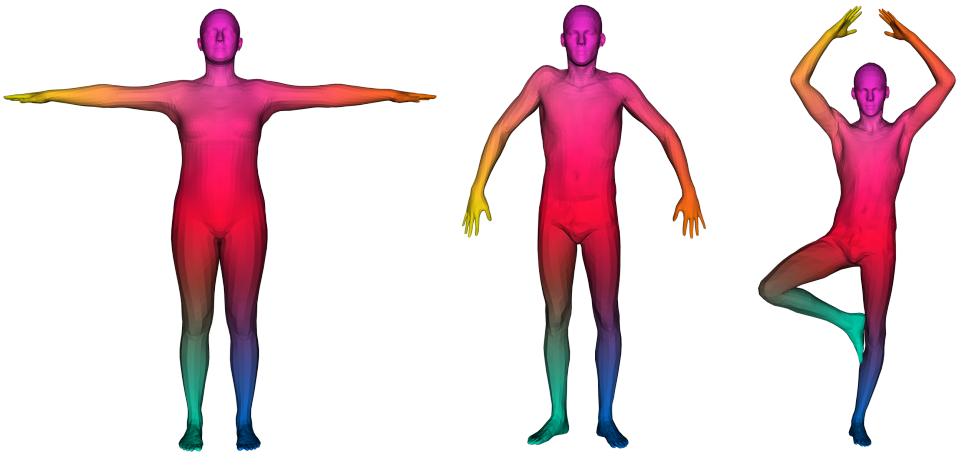


Figure 6.1 Alignment with the template: The 3 meshes in different poses are displayed with different colors on extremities. This validates the choice to work on this particular section of the fiber bundle $\Pi : \mathscr{H} \to \mathscr{H}/G$

6.3 Riemannian Analysis of aligned Human Shapes

Since human surfaces are represented as elements of $\mathscr{S}_0$, it is easy to define Riemannian metrics for them. We will explain in this section in depth the choice of the metric and its meaning.

6.3.1 Elastic Riemannian Metric

Consider a parameterized surface $f : \mathscr{T} \to \mathbb{R}^3$, the regular surface $S = f(\mathscr{T})$, g its first fundamental form, or Riemannian metric on S, and n the Gauss map. We consider the following relationship between parameterized surfaces on one hand and the product space of metrics and normals on the other :

$$\begin{array}{rccc} \Phi : & \mathscr{S}_0 & \longrightarrow & \mathrm{Met}(\mathscr{T}) \times \mathscr{C}^\infty(\mathscr{T}, \mathbb{S}^2) \\ & f & \longmapsto & (g, n). \end{array} \tag{6.1}$$

Thanks to the fundamental theorem of surfaces, we do not lose any information about the shape of a surface f if we represent it by the pair (g, n). The induced metric g captures the intrinsic shape, while the normal n captures the extrinsic geometry of the shape.

6.3.2 The Manifold of Metrics on $\mathscr{T}$ and its Geodesic Distance

The space of first fundamental forms, i.e. positive-definite 2×2 symmetric matrices fields g on $\mathscr{T}$ will be denoted by $\mathrm{Met}(\mathscr{T})$. We will equip this infinite-dimensional space with a diffeomorphism-invariant Riemannian metric, called the Ebin (or DeWitt) metric (Ebin, 1970; DeWitt, 1967), and simplified in (Su *et al.*, 2020b). The Riemannian metric on the tangent

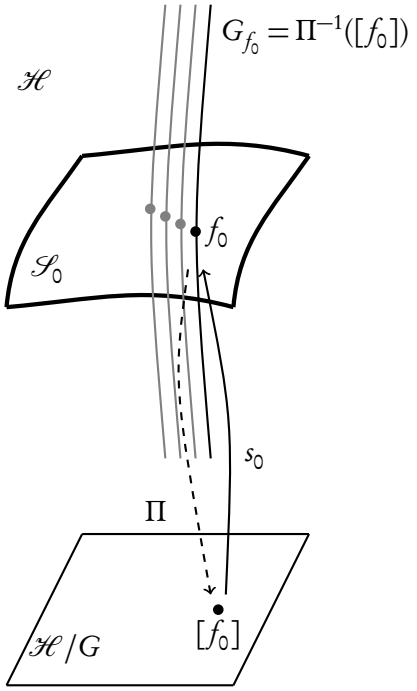


Figure 6.2 Section of the fiber bundle $\Pi : \mathscr{H} \to \mathscr{H}/G$: the one-to-one correspondence with the template mesh allows us to work on the corresponding section $\mathscr{S}_0$ as a shape space. The correspondance initially gives the section for $\mathrm{Diff}^+(\mathscr{T})$, but with Procrustes analysis, the section for SO(3) comes straightforwardly.

space of $\mathrm{Met}(\mathscr{T})$ is defined by:

$$\langle\langle g_h, g_k \rangle\rangle_g = \int_{\mathscr{T}} \mathrm{Tr}(g^{-1} g_{h0} g^{-1} g_{k0}) + \lambda \mathrm{Tr}(g^{-1} g_h) * \mathrm{Tr}(g^{-1} g_k) dA_g \tag{6.2}$$

where $g_{h0} = g_h - \frac{1}{2}\mathrm{Tr}(g^{-1} g_h) g$ is called the traceless part of g_h, and where μ_g denotes the volume form defined by g. The different terms of the metric have nice geometric interpretations. The first term quantifies how the change in first fundamental form of a local surface patch induces shearing (in a mesh, deforming a triangle without changing its volume), and the second term weighted by λ how it induces stretching (scaling a triangle).

The following theorem, from (Su *et al.*, 2020b), presents the geodesic distance between two g_0 and g_1 in (the completion of) $\mathrm{Met}(\mathscr{T})$ for any choice of λ.

Theorem 6.3.1. *Let $g_0, g_1 \in \mathrm{Met}(\mathscr{T})$. The square of the geodesic distance for the family of metrics is*

$$d^{\lambda}(g_0, g_1)^2 = \int_{\mathscr{T}} d^{\lambda,\mathrm{Sym}}(g_0(x), g_1(x))^2 \, dx$$

where

$$d^{\lambda,\mathrm{Sym}}(g_0(x), g_1(x))^2 = 16\lambda\left(s_0^2(x) - 2 s_0(x) s_1(x) \cos(\theta(x)) + s_1^2(x)\right)$$

with

$$
\begin{aligned}
&s_0(x) = \sqrt[4]{\det(g_0(x))}, \quad s_1(x) = \sqrt[4]{\det(g_1(x))} \\
&\theta(x) = \min\left\{\pi, \frac{\sqrt{\lambda^{-1}\operatorname{tr}\left(K_0^2(x)\right)}}{4}\right\} \\
&K(x) = \begin{cases} 0 \text{ if either } g_0(x) \text{ or } g_1(x) \text{ is degenerate} \\ g_0(x)\log\left(g_0(x)^{-1}g_1(x)\right) \text{ else} \end{cases} \\
&K_0(x) = K(x) - \operatorname{tr}\left(g_0^{-1}(x)K(x)\right)g_0(x)
\end{aligned}
$$

Theorem 6.3.2. *Let a, λ, c, three positive real numbers. We equip the space* $\mathrm{Met}(\mathscr{T}) \times C^\infty\left(\mathscr{T}, \mathbb{S}^2\right)$ *with the following Riemannian metric:*

$$
(((\delta g, \delta n), (\delta g, \delta n)))_{g,n} = a\left(\int_{\mathscr{T}} Tr(g^{-1}\delta g\, g^{-1}\delta g) + \lambda Tr(g^{-1}\delta g)^2 \mu_g\right) + c\int_{\mathscr{T}} \langle \delta n, \delta n\rangle dx \tag{6.3}
$$

Let $f_1, f_2 \in \mathscr{S}_0$ and $\Phi(f_1) = (g_1, n_1), \Phi(f_2) = (g_2, n_2)$. Define a distance function $d_{\mathscr{S}_0}$ on $\mathscr{S}_0$ by

$$
d_{\mathscr{S}_0}^{a,\lambda,c}(f_1, f_2) := d(\Phi(f_1), \Phi(f_2)) \tag{6.4}
$$

where d is the geodesic distance in the space $\mathrm{Met}(\mathscr{T}) \times C^\infty\left(\mathscr{T}, \mathbb{S}^2\right)$. *Then the square of the distance $d_{\mathscr{S}_0}$ between f_1 and f_2, with parameters a, λ, c, is given by*

$$
d_{\mathscr{H}}^{a,\lambda,c}(f_1, f_2)^2 = a d^\lambda(g_1, g_2)^2 + c\int_{\mathscr{T}} d_{\mathbb{S}^2}(n_1(x), n_2(x))^2 dx \tag{6.5}
$$

where d^λ is given by Theorem 6.3.1 and $d_{\mathbb{S}^2}(n_1(x), n_2(x)) = \arccos\langle n_1(x), n_2(x)\rangle$ is the geodesic distance on $\mathbb{S}^2$.

We will see in the next chapter the relationship between this family of metrics and the Sobolev metric defined in Chapter 3. In the next chapter, we will indeed work on unparameterized surfaces and need to stabilize computations (to be robust to noise in scans for example), therefore we will use an extended family of metrics deriving from the Sobolev metric. The current metric we work with is however sufficient when we work on aligned human bodies.

6.3.3 Interpolation of shapes as computation of geodesic paths

As mentioned above, an important advantage of our Riemannian approach over many past papers is its ability to compute not only the distance between two human surfaces but also the geodesics or the deformations between shapes. The computation of geodesics requires the minimization of energy. In (Tumpach *et al.*, 2016) the path-straightening method is used to find critical points of the energy functional. Starting with an arbitrary path, the method consists of iteratively deforming (or "straightening") the path in the opposite direction of the gradient, until the path converges to a geodesic. The problem would then be a problem of optimization on the set of vertices of the shape. However, this can lead to numerical instabilities. We will use another, more stable approach (Su *et al.*, 2020a). In this approach, given two shapes f_0 and f_1, after choosing a time step $\frac{1}{T}, T \in \mathbb{N}$, the path is set to the linear path (initialization) on which we add a sum of deformations:

$$
\begin{aligned}
&f(t_0) = f_0, \quad f(t_T) = f_1 \\
&f(t_i) = (1 - t_i)f_0 + t_i f_1 + \sum_j \beta_{ij}\mathscr{D}_j
\end{aligned} \tag{6.6}
$$

Where $\mathscr{D}_j$ is an orthogonal basis of $N_{\mathscr{D}}$ plausible deformations gathered beforehand. The computation of the geodesic requires the minimization of the energy functional $E(\beta)$, defined by:

$$E(\beta) = \int_0^1 \left(\left(\frac{d\Phi(f(t))}{dt}, \frac{d\Phi(f(t))}{dt} \right) \right)_{\Phi(f(t))} dt \tag{6.7}$$

with $\beta \in \mathbb{R}^{(T-2)*N_{\mathscr{D}}}$ the vector containing all β_{ij} presented in equation 6.6, and $((.,.))_{\Phi(f(t))}$ being the pullback by Φ of the Riemannian metric 6.3 on $\mathrm{Met}(\mathscr{T}) \times C^\infty\left(\mathscr{T}, \mathscr{S}^2\right)$.

To find the optimal coefficients α, similar to (Su *et al.*, 2020a), we employ the Broyden Fletcher Goldfarb Shanno (BFGS) method (Fletcher, 1987), implemented in the SciPy library (Virtanen *et al.*, 2020) where we calculate the gradient using the automatic differentiation feature of PyTorch library (Paszke *et al.*, 2019).

Basis Deformations In (Kurtek *et al.*, 2012), (Laga *et al.*, 2017), (Su *et al.*, 2020b), (Tumpach *et al.*, 2016), spherical parameterization of 3D objects is used and spherical harmonics are computed to define the set of deformations. We could use the eigenvectors of the Laplace Beltrami on the template (that generalizes the spherical harmonics to general type of surfaces), but we would require a large number of basis elements to achieve high accuracy and capture all the human surface details. In addition, in the case of human shapes, we are using a human template as a parametrization and there are several publicly available dynamic human shapes that can be used to build a PCA basis of deformations.

In our case to build such real deformations, we use the publicly available Dynamic FAUST dataset (Bogo *et al.*, 2017), which contains motions registered to the template $\mathscr{T}$. 10 individuals (5 males, 5 females) perform 14 different motions, sampled at the rate of 60 frames per second. Given a set of motions, we collect deformations by gathering differences from the sequences. Let $(m_1, ..., m_T) \in \mathscr{S}_0$ be a motion available in the dataset. We define the small deformations that we collect from the motions as the family $(m_{n\tau+\tau} - m_{n\tau})_n$, with τ being a time interval chosen manually, fixed to 10 frames ($\simeq$160 ms). Thus, given a set of training samples, we can compute its PCA basis. In our experiments, the number of PCA basis elements required is of the order of 100.

Note that, by construction, adding a deformations of the basis of deformation to an aligned human shape will not destroy the alignment with the template.

Algorithm 1: Computation of Geodesics

Input: the source and target surfaces f_1 and f_2, a, λ, c the parameter of the elastic metric

Output: f_{geo}: the geodesic connecting f_1 and f_2

1: Initialize $\beta_{ij} = 0$ and $f(t_i)$ by linear path;

2: Define the energy functional $E(\beta)$ in an automatic differentiation framework (PyTorch here), that computes the gradient value $\nabla_\beta E$ along the functional value;

3: Minimize E with respect to β with a BFGS implementation (SciPy *BFGS* or *L-BFGS-B*), that uses the gradient $\nabla_\beta E$;

4: Set the geodesic to be: $\alpha_{\text{geo}}(t_i) = t_i f_0 + (1 - t_i) f_1 + \sum_j \beta_{ij} \mathscr{D}_j$;

5: **return** *the final geodesic* α_{geo}

6.4 Statistical Analysis of Human Shapes

We are interested in defining a notion of "mean" for a given set of human shape. Let $f_1, \dots f_n$ be a set of human shapes. The mean of a set of human shapes is the human shape that is as close as possible to all of the human shapes in the set of human shapes, under the distance metric defined by Equation 6.5. This is known as the Karcher mean and is defined as the human shape that minimizes the sum of squared distances to all of the human shape in the given human shape. In order to find the Karcher mean one can define the following functional:

$$\mathcal{V} : \mathcal{S}_0 \to \mathbb{R}, \mathcal{V}(f) = \sum_{i=1}^{n} d(f_i, f)^2 \tag{6.8}$$

That is differentiable with the distance previously computed. We initialize the Karcher mean as f_1 and set it to be the sum of f_1 with a linear combination of deformations:

$$\tilde{f} = f_1 + \sum_j \beta_j \mathcal{D}_i$$

The functional to minimize becomes:

$$\mathcal{W}(\beta) = \mathcal{V}(f_1 + \sum_i \beta_i \mathcal{D}_i)$$

Algorithm 2: Karcher Mean of Human Shapes

Input: $f_1, \dots f_n$ a set of human body, a, λ, c the parameter of the elastic metric
Output: $\bar{f}$: Karcher mean
1: Initialize $\beta = 0$ and $\tilde{f} = f_1$ by the first shape in the set;
2: Define the Karcher mean functional $\mathcal{W}^{a,\lambda,c}(\beta)$ in an automatic differentiation framework (PyTorch here) that computes the gradient value $\nabla_\beta \mathcal{W}$ along the functional value;
3: Minimize $\mathcal{W}$ with respect to β with a BFGS implementation (SciPy *BFGS* or *L-BFGS-B*), that uses the gradient $\nabla_\beta \mathcal{W}$;
4: Set the Karcher mean to be: $\bar{f} = f_1 + \sum_i \beta_i \mathcal{D}_i$;
5: **return** *Karcher mean*

6.5 Experiments

6.5.1 Assessment of the Family of Elastic Metrics

To further assess the pertinence of the family of elastic distances defined in Equation 6.5 in human shape and pose analysis, we measured pairwise distances of the metric on the registrations present in the FAUST dataset (Bogo *et al.*, 2014). It contains 10 individuals (5 males, 5 females) in 10 different poses. We present in Figure 6.3 2D visualizations of the dataset using the

t-Distributed Stochastic Neighbor Embedding (t-SNE) algorithm (van der Maaten & Hinton, 2008).

In Figure 6.3, left, it is clear that the 3D human with similar poses belong to very close distributions. These results show the assumption that given $a = 0, \lambda = 0, c = 1$ (normal field L_2 metric), the metric is preserved under shape change, and could be used in pose and motion analysis application (Luo *et al.*, 2016; Veinidis *et al.*, 2019) (although, we would need registrations before applying the metric). The Figure 6.3, right, shows that 3D human with similar shape belong to very close distribution. These results states the assumption that given $a = 1, \lambda = 0.0001, c = 0$, the metric is preserved under pose change, and could be used in many shape analysis application approaches (Pickup *et al.*, 2016) and (Lian *et al.*, 2013).

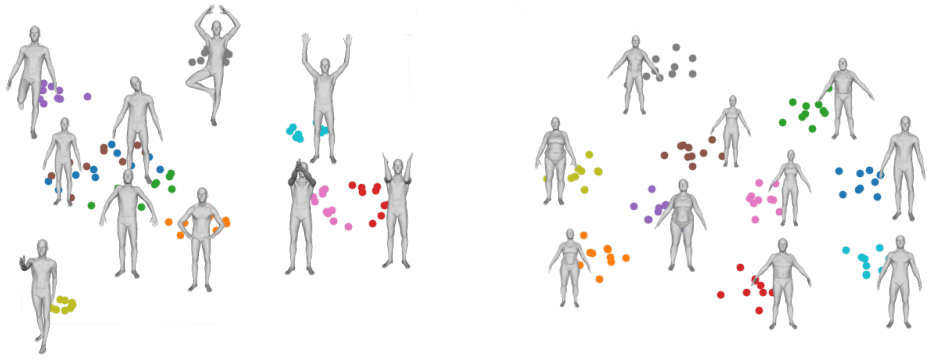


Figure 6.3 2D visualization of the FAUST dataset by our method using t-SNE algorithm based on the metric from equation 6.5. On the left, the metric parameters are set to $a = 0, \lambda = 0, c = 1$. Each color represents a class of pose and a class representative is also displayed. On the right, the metric parameters are set to $a = 1, \lambda = 0.0001, c = 0$. Each color represents a class of shape and a class representative is also displayed.

6.5.2 Geodesics and Karcher Mean

We performed a number of experiments using human surfaces of same and different persons under a variety of pose and shape, and studied the resulting geodesic paths.

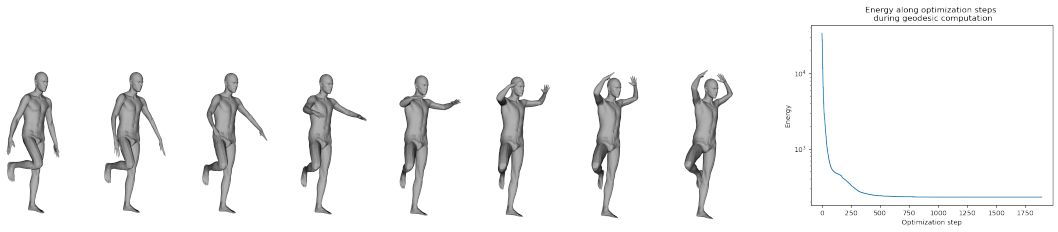
Figure 6.4 shows the geodesic path between f_1 (shown in far left) and f_2 (shown in far right). Drawn in between are human surfaces denoting equally spaced points along the geodesic path. In terms of the Riemannian metric chosen, these paths denote the optimal deformations in going from the first human body to the second and the path lengths quantify the amount of deformations. For this experiment, we also provide a curve of the energy, available right to the paths, which shows that the energy decreases smoothly with time.

For the first path, the change in the pose induces small changes in shape. We thus want to minimize the shape change along the path, which would set the extrinsic parameters $c = 0$. We find that $a = 1, \lambda = 1$ gives the best visual results.

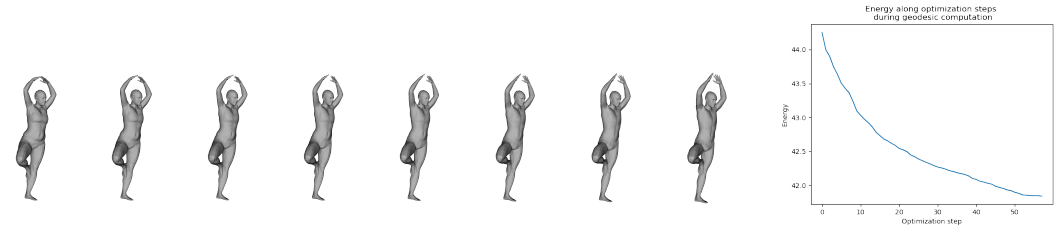
The second path is a path with change in shape. We thus want to minimize the pose change along the path, which would set parameters $a = \lambda = 0$, and the normal parameter $c = 1$.

The geodesic computation were made on a computer setup with Intel(R) Xeon(R) Bronze 3204 CPU @ 1.90GHz, and a Nvidia Quadro RTX 4000 8GB GPU. The computation time of

the different geodesics took less than 5 mins.



(a)



(b)

Figure 6.4 Examples of geodesic path between f_1 and far left and f_2 far right: (a) with metric parameters (a=1, $\lambda = 1$, c=0), (b) with metric parameters (a=0, $\lambda = 0$, c=1). The corresponding energy evolution during optimization are displayed on the right. Computation time was respectively 3min31s and 10.6s.

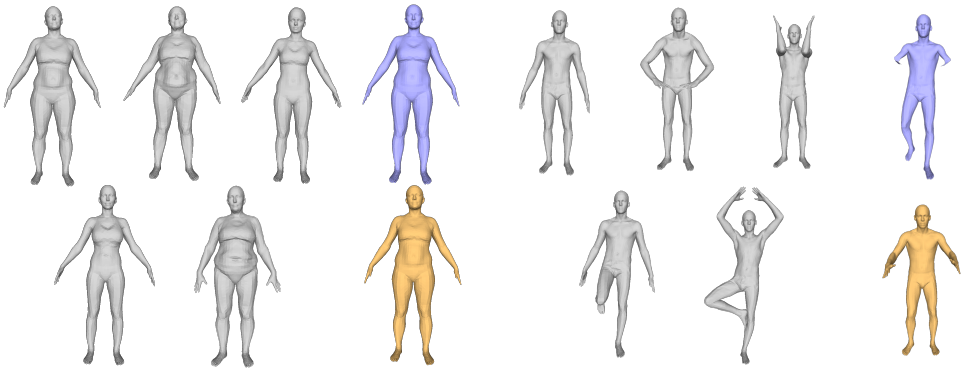
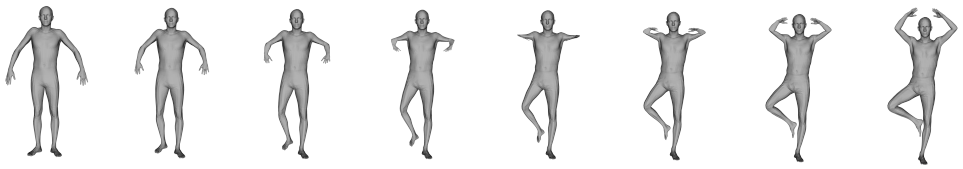


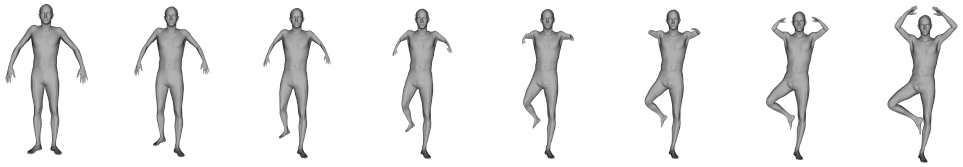
Figure 6.5 Karcher means of shapes. Left: Karcher mean (yellow) for five different people relatively to the distance with metric parameters ($a = 0, \lambda = 0, c = 1$). Right: Karcher mean (yellow) for five different pose relatively to the distance with metric parameters ($a = 1, \lambda = 1, c = 0$). In blue, we display the corresponding linear means.

An example of Karcher mean is shown in Figure 6.5, where five human bodies in the same pose but with different shapes are averaged.

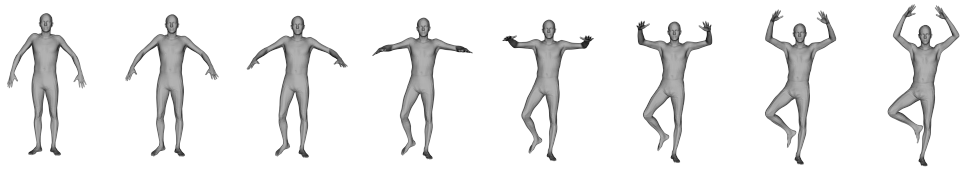
We also compare the results obtained with our method to the results using linear geodesic path, SRNF and SMPL descriptors.



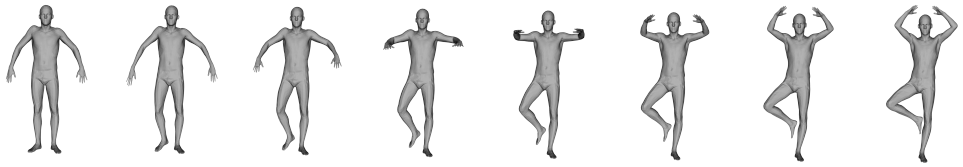
(a) Linear geodesic path



(b) Geodesic computed with SRNF



(c) Geodesic computed in SMPL space



(d) Geodesic computed with our approach, metric parameters are set to $a = 1, \lambda = 1, c = 0$. Computation time was 3min10s.

Figure 6.6 Comparison of our approach with different frameworks. We observe that the linear initial path is challenging, while the SRNF path induces distortion in the shape. Finally although the SMPL geodesic is able to keep the shape, we argue that the path of our approach is the most natural path compared to the one proposed by SMPL: the natural deformations between the source and target shape would indeed bend more the elbow.

1. The linear path defined by:

$$\begin{aligned} f(t_0) &= f_0, \quad f(t_T) = f_1 \\ f(t_i) &= (1-t_i) f_0 + t_i f_1 \end{aligned} \tag{6.9}$$

2. The SRNF geodesic path is also visualized. This representation has been used to analyze human shapes with interesting results (Laga *et al.*, 2017; Su *et al.*, 2020a). The SRNF is a pointwise representation based on $q = \sqrt{A}n$, where $A = ||f_u \times f_v||$ is the area, and n the normal field. We compute the geodesic for the $SRNF$ representation with the same method as presented in this section.

   As shown in Figures 6.6(a) and (b), the linear interpolation and SRNF lead to unnatural deformations for human paths. The deformation between surfaces contains many artifacts and degeneracies.

3. SMPL body model (Loper *et al.*, 2015) : After fitting SMPL model to the FAUST dataset, we can compute the corresponding geodesic, using the resulting shapes of the linear path in the SMPL parameter space, see Figure 6.6. While the deformation propose by the SMPL body model is in some way plausible, we first argue that the pose deformation proposed by SMPL does not bend enough the elbow: this is due to the linear interpolation of the elbow joint angle. In addition, one can observe that the target and sources shapes are slightly different than for our shapes: this is due to the fitting step of SMPL: the resulting shape is the closest shape with plausible SMPL parameters, not exactly the input shape.

In both examples, our approach provides better results.

6.6 Application to Pose and Shape Retrieval

Here, we demonstrate how the proposed metric can be exploited for 3D human retrieval. Given a 3D human, we look for the similar 3D human in a database.

6.6.1 Evaluation and comparison

We test the usefulness of the family of metrics (Equation 6.5) in 3D human shape and pose retrieval. We use Nearest neighbor (NN), First-tier (FT), Second-tier (ST) as evaluations criteria.

Comparisons: We propose several methods for comparison with our method.

- We gather the extrinsic latent vectors, which lives in $\mathbb{R}^{12}$, along with their intrinsic latent vectors of GDVAE (Aumentado-Armstrong *et al.*, 2019) trained netwrok. We use them for human pose retrieval and shape retrieval respectively.
- We gather pose and shape latent vector of (Zhou *et al.*, 2020a) autoencoder, which lives in $\mathbb{R}^{112}$ for pose and shape retrieval respectively
- The best descriptor results of (Pickup *et al.*, 2016) is presented, won by the Area Projection Transform (Giachetti & Lovato, 2012), for human shape retrieval

- We compare to the SRNF distance that showed reliable results for pose retrieval
- we use both shape and pose representation from the SMPL body model for the respective retrieval tasks.

6.6.2 Experimental results

In this section, we perform evaluations of our method in FAUST dataset. We evaluate on pose and shape retrieval. The evaluation results in Table 6.1 demonstrate that our method outperforms the previous state of the art shape retrieval methods in term of NN criteria. The Table 6.2 shows that the proposed approach provides the best results on pose retrieval in term of FT and ST criteria. We also find that for shape retrieval, the best parameters are $a = 1, \lambda << a$. The computation times for each pairwise distance were $\simeq$70 ms and $\simeq$80 ms for pose and shape retrieval respectively.

Repr.	NN	FT	ST
GDVAE intrinsic	27	24.8	46.2
Zhou et al. shape	42	24.8	42.8
SMPL shape vector	98	72.4	86.7
APT	96	86.5	96.2
Metric (1, 0.0001, 0)	**100**	**94.8**	**97.1**

Table 6.1 FAUST dataset results for shape retrieval

Repr.	NN	FT	ST
GDVAE extrinsic	60	38.0	54.2
Zhou et al. pose	82	69.2	83.4
SMPL pose vector	80	84.4	95.2
SRNF	73	77.7	94.4
Metric (0, 0, 1)	**85**	**88.3**	**97.6**

Table 6.2 FAUST dataset results for pose retrieval

6.7 Conclusion

In this Chapter we proposed a novel Riemannian framework which allows not only to compute a metric between human bodies under pose and shape changes, but also provides a geodesic path between human bodies, and statistical tools (eg. mean of human shape). We have demonstrated the utility of the proposed framework in pose and shape retrieval of human body. The main limitation of our method lies in the requirement of a template. In the next Chapter, we will extend our approach and propose a way to solve this specific problem.

Chapter 7

BaRe-ESA: A Riemannian Framework for Unregistered Human Body Shapes

Contents

7.1 Introduction

In the last Chapter, we proposed a Riemannian framework that allows for several applications such as shape interpolation and human shape retrieval. This framework is in practice limited to registered human bodies. However being able to do shape interpolation, and more, like shape extrapolation, – the task of finding a plausible deformation given a body scan and a corresponding initial deformation (movement) – and random shape generation – directly on human body scans opens for a broader range of possibilities. Furthermore, one is interested in obtaining a natural entanglement of changes in the pose and in the shape of the human body (Cosmo *et al.*, 2020), which allows the developing of algorithms for operations such as motion transfer (Basset *et al.*, 2021).

7.1.1 Differences with other common approaches

Our method is different from traditional approaches composed of separated registration and deformation step because as we explained in Chapter 2, significant bias can be introduced during the registration step. We thus aim for a unified approach by introducing a new pipeline that quantifies geometric differences between unregistered human body scans, i.e., that does not require prior point correspondences or consistent parametrization across the dataset. Furthermore, we are not only interested in a pure metric comparison of two individuals, but also in estimating plausible deformation processes from one human body to the other. To that end, we introduce a transformation model that allows disentangling changes in the pose and in the shape of the human body so as to obtain realistic ways to interpolate from one scan to another, extrapolate a given motion or transpose it to a new individual and even to generate visually plausible random pose and shapes; see Figure 7.1 for a schematic vizualization of our framework.

Riemmanian deformation of shapes (Srivastava & Klassen, 2016; Jermyn *et al.*, 2017) can be done in an unparametrized way using elastic metrics such as in the previous Chapter because they are invariant to the parameterization of surfaces. In the field, it is generally assumed that the surfaces are given by analytic (spherical) representations. More than the (already hard) problem stated by the spherical parameterization itself, resumed in the last chapter, in the context of real data (raw body scans) inducing imaging noise in particular, the parameterization problem itself becomes a highly non-trivial task, that can introduce significant bias and error in subsequent statistical analyses.

As we saw in Chapter 5, the varifold representation of surfaces (Charon & Trouvé, 2013; Kaltenmark *et al.*, 2017), can be used to define discrepancy metrics between shapes that are not only invariant to parameterization but also robust to scan inconsistencies such as potential noise or holes. The approach of this chapter follows some of the recent advances (Bauer *et al.*, 2021b; Hartman *et al.*, 2022) that combine Riemannian elastic metrics with varifolds discrepancy terms and thereby overcome the aforementioned difficulties. Therefore, our approach does not require having a consistent mesh structure across the dataset and performs well on human body scans with different numbers of vertices and even under the presence of topological noise (e.g. holes in the scans).

Following the success of the basis-driven deformation approach of the previous chapter, our method is closely related to data-driven **human body latent spaces**, where one defines

deformations as linear pathq in the latent space that are decoded with a nonlinear neural network into human shapes.

In contrast, in our proposed pipeline, we impose an affine map (basis restricted deformations) from a given low dimensional latent space to a corresponding space of human body transformations, which is built on the use of some pre-estimated bases to represent infinitesimal body shape and body pose deformations. However, unlike the aforementioned deep learning architectures, we do not rely on the standard Euclidean metric in the latent space but instead on the pullback of a second-order parametrization-invariant Sobolev metric in the space of the deformation field. Thus, interpolating paths in latent space are no more straight lines, but are associated with (sub-Riemannian) geodesics in the surface space for this metric. One important feature of our method is that it requires a relatively small dataset and minimal time to train but at the expense of solving a non-trivial optimization problem at the encoding stage. Yet, as our results will show, the specific Riemannian structure imposed on the latent space leads to better interpretability and generalization properties.

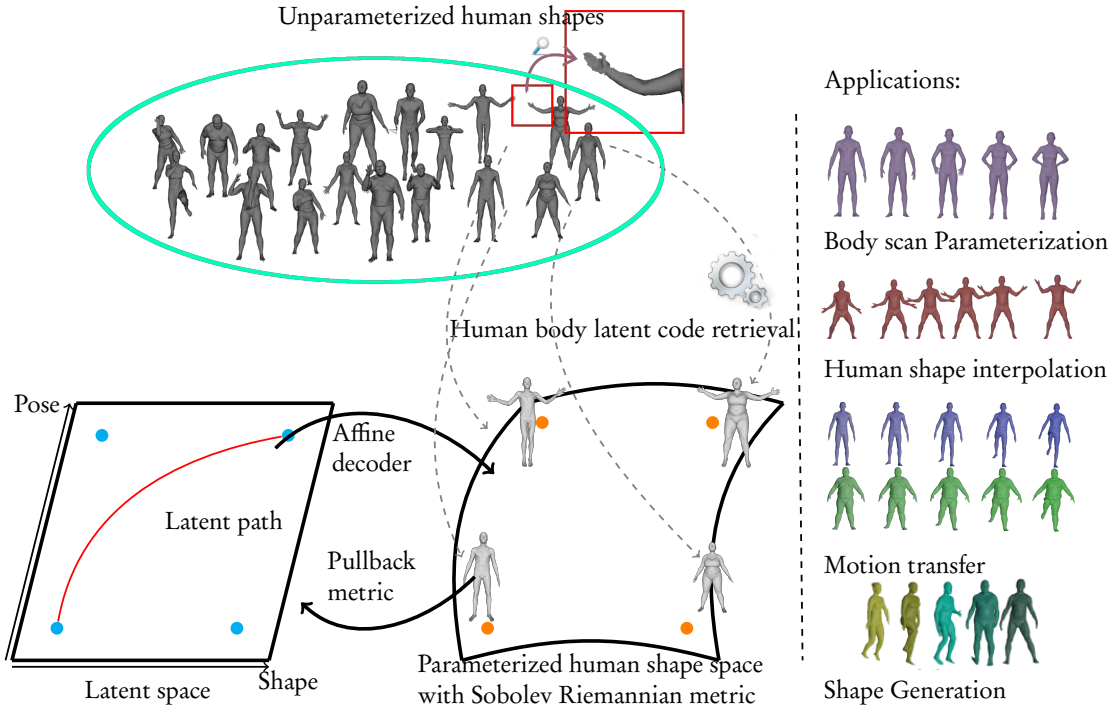


Figure 7.1 Overwiew of our method. We seek to represent unparameterized human bodies, with different mesh connectivity, and possible noise or topological changes in a disentangled latent space. We define our latent space as the sum of pose and shape spaces. The paths in the latent space are not linear but curved, corresponding to geodesics in the parameterized human body space. After retrieving the latent codes of the human bodies, we can use the space along with its Riemmanian metric to solve several problems in human body deformation: inter-extrapolation, motion transfer, and shape generation.

7.2 A latent space model of human shapes

7.2.1 Shape Space of Human Bodies

Working on the space of human shapes: Instead of what was done in the previous chapter, we do not restrict ourselves to a section of the surface space of registered human shapes but work on the shape space as defined in chapter 3. We suppose in this chapter that translations and rotations are filtered out during a preprocessing step, and focus only on the parameterization group $\mathrm{Diff}^+(\mathscr{T})$, i.e our shape space is $\mathscr{H} = \mathscr{S}/\mathrm{Diff}^+(\mathscr{T})$. We equip our shape space with a Riemannian metric, in a similar process to the previous Chapter. The problem of interpolation and extrapolation is thus still a problem of geodesic computation.

7.2.2 Elastic Riemannian metric

Following the success of the family of Dewitt metrics in last Chapter, we choose an extension of this family, the second-order metrics with six parameters proposed in (Hartman *et al.*, 2022). They derive from the Sobolev metric briefly mentioned in chapter 3:

$$((h,k))_f = \int_{\mathscr{T}} \langle h,h\rangle + \mathrm{tr}(dh(p)(g_f(p)^T g_f(p))^{-1} dk(p)^T) + \langle \Delta_f h, \Delta_f k\rangle \,\mathrm{vol}_f \tag{7.1}$$

We use the construction of (Su *et al.*, 2020b) that divides the Sobolev first-order terms into four orthogonal terms (relative to the metric g^{-1}). Equation (7.1) is decomposed using real-valued weighting coefficients producing a six-parameter family of metrics given by

$$\begin{aligned}((h,k))_f = \int_{\mathscr{T}} \Big(& a_0\langle h,k\rangle + a_1 g_q^{-1}(dh_m, dk_m) \\ & + b_1 g_f^{-1}(dh_+, dk_+) + c_1 g_f^{-1}(dh_\perp, dk_\perp) \\ & + d_1 g_q^{-1}(dh_0, dk_0) + a_2\langle \Delta_f h, \Delta_f k\rangle \Big)\,\mathrm{vol}_q\,.\end{aligned} \tag{7.2}$$

Each of these terms has natural geometric interpretations. The second-order term weighted by a_2 penalizes tangent vectors that increase the local curvature of the surface. It thus provides enough regularity to prevent the occurrence of singularities along the solution of the interpolation and extrapolation problem described above. The zeroth-order term weighted by a_0 penalizes how far the surface is moved weighted by the volume form of the surface. The terms weighted by a_1, b_1 and c_1 corresponds to the one weighted by a, λ, and c in the previous chapters, and measure how a mesh triangle shears, stretches, and bends. The term d_1 is the term orthogonal to the last three terms, measuring how much the tangent space frame is rotated (rotation of a triangle around the normal axis).

Following the experiments of the last Chapter, we choose a_1 and b_1 to be the largest coefficients, but this time, we use all coefficients that will serve as regularizers for the output shapes in order to be robust to variations in scan quality, following (Hartman *et al.*, 2022).

7.2.3 Latent space model

We extend our low-dimensional approach in the previous chapter, into a latent space model. We consider human bodies as a combination of a certain set of admissible deformations applied

to a template shape. Therefore in our model, we restrict ourselves to surfaces f that can be written as linear combinations of a basis of body motion deformations, $\{h_i\}_{i=1}^m$, and a basis of shape deformations, $\{k_i\}_{i=1}^n$. Thus, all of the shapes we consider are determined by the latent space $\mathscr{L} \subset \mathbb{R}^{n+m}$ of coefficients for the combined basis. This latent space model is thus related to the space of surfaces by the mapping $F : \mathscr{L} \to \mathscr{S}$ via

$$F(\alpha^j) = \bar{f} + \sum_{i=1}^{m} \beta_i^j s_i + \sum_{i=m+1}^{m+n} \alpha_i^j r_{i-m}.$$

where $\bar{f}$ is our parameterized template human body shape, s_i is a basis of body shape deformations, and where r_i is a basis of body pose deformations. The basis choice is a crucial ingredient for the performance of the proposed method. We construct our basis in a similar way as in the previous chapter, taking profit from the available training data. The details of the construction are available in Section 7.3.3. We then equip our latent space with the pullback of the second-order metric from the surface space via F.

7.3 Riemannian analysis of human body scans

7.3.1 Interpolation as a geodesic path computation

Let f_0, f_1, two unparametrized human body scans in an arbitrary given parameterization. As said in Chapter 3, the geodesic between two shapes f_0 and f_1 is the solution to :

$$\inf_{\gamma \in \mathrm{Diff}^+(\mathscr{T})} \inf_{\alpha \in \mathscr{P}_{f_1}^{f_2 \circ \gamma}} \int_0^1 \langle \partial_t \alpha(t), \partial_t \alpha(t) \rangle_{\alpha(t)} dt$$

Where α is now a path between shapes' latent codes. Interpolating thus requires minimizing the geodesic energy over all paths $\alpha : [0,1] \to \mathscr{L}$ and diffeomorphisms $\gamma \in \mathrm{Diff}^+(\mathscr{T})$ such that $F(\alpha(0)) = f_0$ and $F(\alpha(1)) = f_1 \circ \varphi^{-1}$. This problem is much more complicated than the previous chapter. First, one needs to model the action of the diffeomorphism group on the two endpoints. Moreover, we need to take in account the fact that raw body scans are, in general, not equipped with a consistent mesh structure, i.e., different scans can have different resolutions, different mesh structures, or even admit imaging errors and noise such as holes or missing parts, as seen in Figure 7.1.

To circumvent these difficulties we instead consider a relaxed formulation of the geodesic problem given by the following energy

$$\tilde{E}(\alpha) = \int_0^1 \overline{G}_\alpha(\partial_t \alpha, \partial_t \alpha) dt + \lambda \rho(F(\alpha)(0), f_0) + \lambda \rho(F(\alpha)(1), f_1), \tag{7.3}$$

where ρ is a reparametrization blind similarity measure, i.e., ρ satisfies the fundamental property $\rho(f_0, f_1) = 0$ if and only if f_0 and f_1 represent the same shape, i.e., there exists a $\varphi \in \mathrm{Diff}^+(\mathscr{T})$ such that $f_0 = f_1 \circ \varphi^{-1}$. We have thus reduced the geodesic boundary value problem to an unconstrained minimization problem over all paths $\alpha : [0,1] \to \mathscr{L}$.

There are various different possibilities for the similarity term, such as the Hausdorff or Chamfer distance. We will rely on the kernel metrics on the space of varifolds, which were

presented in detail in Chapter 4. An important advantage of this framework in this case is the fact that the resulting fidelity loss function remains differentiable with respect to the positions of the shapes' vertices.

We remind that after discretization, the varifold distance does not require the given meshes to have a consistent mesh structure which allows us to compare human body scans with different numbers of vertices and even different topologies, e.g. in the case of the presence of holes in the scans. We denote by σ the spatial support of the metric, and will use the *current* kernel for the spherical kernel, which is known to have better behavior on noisy shapes (Kaltenmark *et al.*, 2017).

Using the varifold metric reduces the infinite-dimensional minimization problem to a finite-dimensional one. The variables are the latent space coefficients of the path. We again use a scipy BFGS implementation, following common approaches (Hartman *et al.*, 2022).

In practice finding the geodesic between two shapes is equivalent to finding the corresponding latent codes of the two shapes during the energy minimization. We found it more efficient to separately retrieve the latent codes using this approach before performing the interpolation in the latent space. Given a human body scan, one can define the latent code retrieval problem as shooting a geodesic that starts from the template and ends at a shape that has zero varifold distance to the scan. We adopt a multiscale approach for the varifold metric, in a similar way as (Hartman *et al.*, 2022) where sequential minimizations are performed by decreasing σ of the varifold term and increasing the balancing term λ. To improve efficiency and avoid local minima, we add an initialization term based on training latent codes: we initialize the latent code to be the one that has the closest corresponding varifold distance to the input scan.

We describe the algorithm of this procedure in Algorithm 3.

7.3.2 Extrapolation as geodesic shooting

The shape extrapolation problem consists in predicting the future evolution of a surface (human body) given an initial deformation direction. In our Riemannian framework this reduces to solving the geodesic equation with given initial condition $q(0) = q_0$ (the initial pose) and $\partial_t q(0) = h$ (the direction of deformation). The geodesic equation is the first-order optimality condition of the energy functional; it is non-linear PDE, that is second order in time t and fourth order in space. For the exact formula of this equation, which is rather lengthy and not particularly insightful, we refer the interested reader to the literature, see eg. (Bauer *et al.*, 2011). To solve such initial value problems in our latent space, we modify methods of discrete geodesic calculus (Rumpf & Wirth, 2013) for our setting. We approximate the geodesic starting at α^0 in the direction of β with a PL path with $N+1$ evenly spaced breakpoints. At the first step, we set $\alpha^1 = \alpha^0 + \frac{1}{N}\beta$ and find α^2 such that $F(\alpha^1)$ is the geodesic midpoint of $F(\alpha^0)$ and $F(\alpha^2)$, i.e., we solve for α^2 such that

$$\alpha^1 = \operatorname*{argmin}_{\tilde{\alpha}} \left[\left((\beta^0, \beta^0) \right)_{\alpha_0} + \left(\left(\tilde{\beta}, \tilde{\beta} \right) \right)_{\tilde{\alpha}} \right]$$

where $\beta_0 = \tilde{\alpha} - \alpha^0$ and $\tilde{\beta} = \alpha^2 - \tilde{\alpha}$. Differentiating with respect to $\tilde{\alpha}$ and evaluating the expression to minimize at α^1, and expressing the result component by component (over our

Algorithm 3: Latent code retrieval of a scan

Input: The target scans f_0;
$a_0, a_1, b_1 c_1, d_1, a_2$ the parameter of the Sobolev elastic metric;
$(\lambda_k, \sigma_k)_{k=0}^{p}$ the balancing weight and the spatial support of the varifold distance at each refinement step

Output: f_{geo}: the geodesic connecting the template shape $\bar{f}$ to the representative $[f_0]$ in the template space and the corresponding coefficient path α in the latent space.

Initialize $\alpha_{ij} = 0$ and the path as $\bar{f} + \sum_{i=1}^{m} \alpha_i^j h_i + \sum_{i=m+1}^{m+n} \alpha_i^j k_{i-m}$. ;

for $k \leftarrow 0$ **to** p **do**

Define the energy functional $E(\alpha) = \int_0^1 \left(\left(\frac{dF(\alpha)}{dt}, \frac{dF(\alpha)}{dt} \right) \right) dt + \lambda_k \Gamma(F(\alpha(1)), f_0)$ in an automatic differentiation framework (PyTorch here), that computes the gradient value $\nabla_\alpha E$ along the functional value;

Minimize E with respect to α with a gradient descent algorithm (SciPy *BFGS* or *L-BFGS-B*), outputing optimal α_{out} coefficients based on initialization α;

Set $\alpha = \alpha_{out}$;

end

Set $[f_0]$ to be the endpoint of the final geodesic;

return α, f_{geo}*and* $[f_0]$

two basis of deformations $\{r_i, s_i\}$), we obtain the system of equations

$$\begin{aligned} &2\left(\left(\beta^0, h_i\right)\right)_{\alpha_0} - 2\left(\left(\tilde{\beta}, s_i\right)\right)_{\alpha_1} + \left\{D_{\alpha^1}\left[\left(\left(\tilde{\beta}, \tilde{\beta}\right)\right)_{\cdot}\right]\right\}_i, \\ &2\left(\left(\beta^0, k_i\right)\right)_{\alpha_0} - 2\left(\left(\tilde{\beta}, r_i\right)\right)_{\alpha_1} + \left\{D_{\alpha^1}\left[\left(\left(\tilde{\beta}, \tilde{\beta}\right)\right)_{\cdot}\right]\right\}_{i+m} \end{aligned} \tag{7.4}$$

where the term $\left\{D_{\alpha^1}\left[\left((\beta, \beta)\right)_{\cdot}\right]\right\}_i$, is the i-th component of the gradient of the term $((\beta, \beta))_{\tilde{\alpha}}$ relatively to $\tilde{\alpha}$ (where β is constant), evaluated in α_1. To see general derivations of the problem (where there is no basis for deformations), one can look at (Hartman *et al.*, 2022).

We denote the system of equations in (7.4) by $\Phi(\alpha^2; \alpha^1, \alpha^0) = 0$, where we stress again that α^0 and α^1 are here fixed and known. We solve this system of equations for α^2 using a nonlinear least squares approach, i.e., by computing α^2 as the minimizer of the residual of the system :

$$\alpha^2 = \underset{\tilde{\alpha}}{\operatorname{argmin}} ||\Phi(\tilde{\alpha}; \alpha^1, \alpha^0)||_2^2.$$

We repeat this process $N-1$ times, thereby constructing the discrete solution up to time $t = 1$.

7.3.3 A data-driven basis of deformations

To construct the bases of movements and body type deformations we interpret registered mesh sequences of motions as paths in shape space whose tangent vectors are implicitly restricted to the space of valid motions of a human body. The tangent vectors of those sequences are used as training data, on which we perform principle component analysis to obtain a tractable

Algorithm 4: Motion extrapolation

Input: The input shape q_0, and its latent code α_0;
The input velocity h and the number of steps N ;
$a_0, a_1, b_1 c_1, d_1, a_2$ the parameter of the Sobolev elastic metric;
Output: f_{geo}: the geodesic starting from q_0 with initial velocity h, and the corresponding latent coefficients $\alpha_0, ..., \alpha_{N-1}$
Initialize $\alpha_1 = \alpha_0 + \alpha_h$;
for $k \leftarrow 2$ **to** $N-1$ **do**
  Define the residual function of the PDE system $\Phi(\alpha; \alpha^{t-1}, \alpha^{t-2})$ in an automatic differentiation framework (PyTorch here), that computes the gradient value $\nabla_\alpha \Phi$ along the functional value;
  Minimize Φ with respect to α with a gradient descent algorithm (SciPy *BFGS* or *L-BFGS-B*), outputing optimal α_{out} coefficients based on initialization $\alpha = \alpha^{t-1} + \frac{1}{N} * (\alpha^{t-1} - \alpha^{t-2}$;
  Set $\alpha^t = \alpha_{out}$;
end
Set $f_{\text{geo}} = [f_t]$ to be the shapes corresponding to the final geodesic;
return α *and* f_{geo}

yet expressive basis for the valid pose deformations of a human body. We then collect meshes of the same pose from each identity (generally available in T or A-pose) and compute the (unrestricted) pairwise geodesics between these meshes with respect to our second-order Sobolev metric, where we use the Pytorch implementation of (Hartman *et al.*, 2022)[1]. Note that these meshes show only moderate deformations and thus there are no difficulties with applying the unrestricted matching algorithm. We then collect the tangent vectors to these paths and perform again PCA to define our basis of shape deformations.

Parameter selection: We set experimentally the body pose size to $n = 130$ elements and the shape basis size to $m = 40$ elements. The coefficients for the H^2-metric were chosen in a similar way as in the previous chapter. Furthermore, we added a small coefficient to the second-order term to further regularize the deformations. The final six parameters for the H^2-metric are set to $(1, 1000, 100, 1, 1, 1)$ During the sequential minimizations where the parameter σ of the varifold term is decreased from .4 to .025 and the balancing term λ is increased from 10^2 to 10^8.

7.4 Results

In this section we demonstrate the accuracy of our framework in five different experiments: the registration of human body scans, the interpolation and extrapolation of human body motions, random shape generation and motion transfer.

[1] https://github.com/emmanuel-hartman/H2_SurfaceMatch

7.4.1 Datasets

Training dataset: To construct our basis we will make use of Dynamic FAUST (D-FAUST) (Bogo *et al.*, 2017) dataset. We use the registrations as a training set, and a set of 7 long range sequence of scans, divided in 10 representative mini-sequences, are left for testing.
Test dataset: In addition to these 7 sequences from D-FAUST we tested our algorithm on the static FAUST dataset (Bogo *et al.*, 2014). We selected the 9 unregistered poses of the training set that show no rotations, and use them as a testing set.

7.4.2 Evaluation and comparison

Before presenting our results, we first discuss the procedure that we use to evaluate and compare the different methods. With the first three experiments (registration, interpolation and extrapolation), we perform a thorough comparison to other state-of-the-art approaches for human body analysis that rely on latent space learning. To keep the section more focused, we deliberately restrict to those, and do not consider other methods that can potentially tackle the same tasks but without a low dimensional latent space (Eisenberger *et al.*, 2021), or that are specifically designed for other tasks (Muralikrishnan *et al.*, 2022).

Thus we compare our approach to the following:

- Learning Latent Shape Representations with Metric Preservation (LIMP) (Cosmo *et al.*, 2020) LIMP is a deep learning method modeling shape deformations based a variational autoencoder with geodesic constraints (in the training loss). The encoder part is a PointNet architecture, which makes it invariant to parameterization.

- The ARAPReg method (Huang *et al.*, 2021). ARAPReg models deformations by using what authors call an "auto decoder": deformations are represented by a vector in a latent space, which is then passed through a graph neural network and reconstructs the human shape. The network is regularized using the ARAP energy. In their framework, the latent vectors are recovered in a registered setting using the L^2 metric, i.e., a consistent mesh structure and known-point correspondences are assumed. To make this framework viable for our application we replace the L^2-metric by the varifold distance thereby extending ARAPReg to unregistered point clouds.

We trained both networks on the D-FAUST dataset and reported training details from the respective papers.

In order to evaluate the different methods we will measure the distance between the proposed outputs of different methods and ground truth scans. Note that we only want to evaluate the *quality* of reconstruction, and not solve the matching problem. As the outputs will not be in the same mesh connectivity as the ground truth, we will evaluate them against ground truth using three different remeshing invariant similarity measures:

- the widely used Hausdorff distance, which between two shapes $[f_0]$ and $[f_1]$, is given by the formula:

$$d_H([f_0],[f_1]) = \max\Big(\sup_{x\in[q_0]}(d(x,[f_1])),\ \sup_{y\in[f_1]}(d([f_0],y))\Big)$$

We use the approximate implementation provided by libigl (Jacobson *et al.*, 2018). While the Hausdorff distance provides in general a good insight for the quality of a mesh reconstruction, it can be sensitive to single outliers present in low quality scans.

- As a second measure we consider the Chamfer distance (Fan *et al.*, 2017; Groueix *et al.*, 2018) which can be computed between two point clouds via:

$$d([f_0],[f_1]) = \frac{1}{N_0}\sum_{x_0\in f_0} d(x_0,f_1) + \frac{1}{N_1}\sum_{x_1\in f_1} d(x_1,f_0).$$

We use the Pytorch implementation of Thibault Groueix[2].

- As a final measure of reconstruction quality we will employ the varifold distance with a low bandwith ($\sigma = 0.025$).

7.4.3 Latent code retrieval of human body scans

Given a human body scan f_0 with arbitrary mesh structure, we retrieve the latent code that corresponds to its shape class $[f_0]$ by performing a relaxed geodesic matching problem between $\overline{f}$ and f_0 with T time steps. This produces a geodesic path $\alpha_0,...,\alpha_T$ from the template to the shape class of the target mesh, thus the endpoint α^T is the latent code corresponding to the shape class of f_0.

	Hausdorff	Varifold	Chamfer
LIMP	0.23	0.073	0.098
ARAPReg	0.11	0.021	0.117
BaRe-ESA	**0.08**	**0.014**	**0.019**

Table 7.1 Shape registration results. The training is performed on D-FAUST, and the registered shapes comes from FAUST. We evaluate the performance of each method by measuring the distance from the registered meshes to scans.

We test our procedure on the 90 meshes from the unregistered FAUST dataset (recall that this dataset was not contained in the training process). We also retrieve latent code representations of LIMP and ARAPreg and measure the distance from the reconstructed meshes to the original scans using the evaluation methods outlined in Section 7.4.2. In Figure 7.2 we present a qualitative comparison of the obtained results. A quantitative comparison of the performance of the different methods is presented in Table 7.1 where we demonstrate that BaRe-ESA significantly outperforms the mesh autoencoder methods with respect to all three evaluation metrics.

7.4.4 Interpolation

The interpolation problem is defined as the task of constructing a deformation between two different human body poses, that follows a *realistic* motion pattern. In our Riemannian setup, as described previously, we need to minimize the discrete energy given in (7.3). We use the start

[2] https://github.com/ThibaultGROUEIX/ChamferDistancePytorch

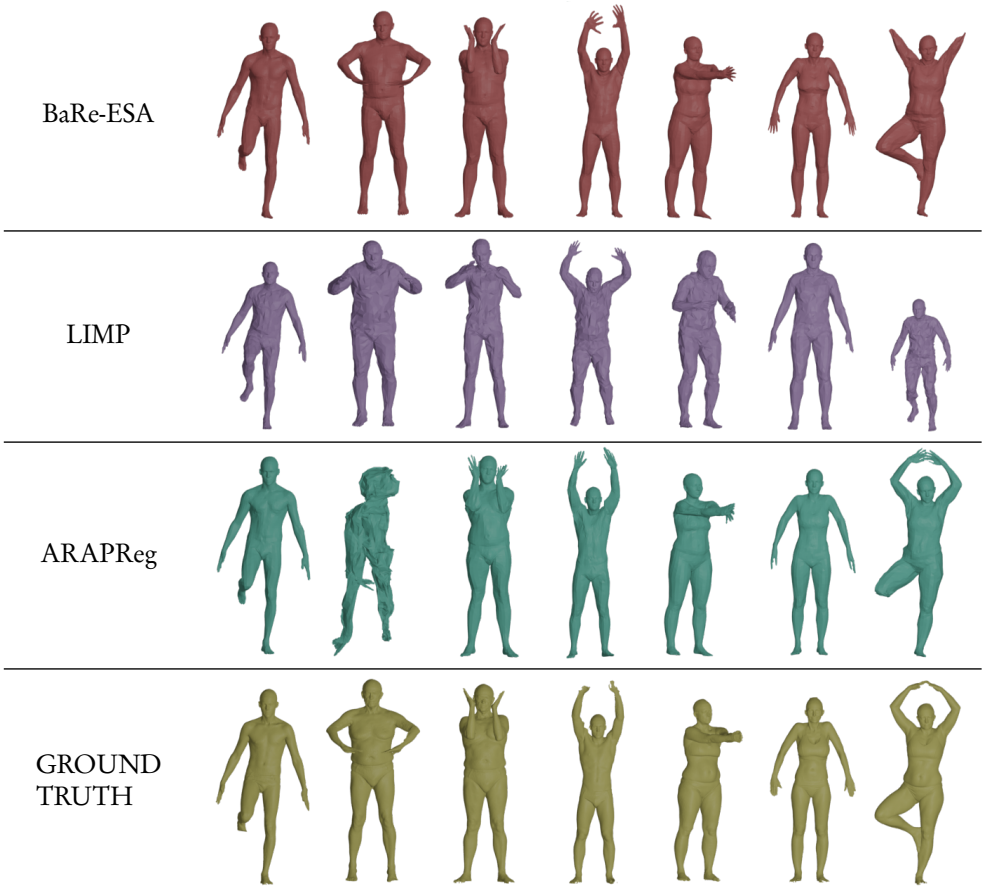


Figure 7.2 Registration of five elements of FAUST using three methods trained on D-FAUST. Both LIMP and ARAPreg fail to generalize to the new data and do not represent the ground truth in certain examples. BaRe-ESA consistently produces a decent representation in all examples, with a failure case displayed in the last column.

	Hausdorff			Chamfer			Varifold		
	LIMP	ARAPReg	BaRe-ESA	LIMP	ARAPReg	BaRe-ESA	LIMP	ARAPReg	BaRe-ESA
punching	0.13	0.12	**0.081**	0.029	**0.020**	0.04	0.060	0.034	**0.022**
running on spot	0.28	0.14	**0.076**	0.112	0.080	**0.069**	0.072	0.052	**0.025**
running on spot b	0.20	0.13	**0.082**	0.125	0.101	**0.045**	0.068	0.040	**0.025**
shake arms	0.34	0.33	**0.078**	0.063	**0.049**	0.076	0.061	0.031	**0.025**
chicken wings	0.18	0.15	**0.093**	0.101	0.083	**0.04**	0.062	0.029	**0.016**
knees	0.060	0.35	**0.051**	0.182	0.266	**0.036**	0.097	0.066	**0.016**
knees b	0.52	**0.054**	0.084	0.168	0.107	**0.029**	0.067	0.019	**0.016**
jumping jacks	0.12	0.32	**0.046**	0.074	0.054	**0.014**	0.070	0.044	**0.015**
jumping jacks b	0.40	0.11	**0.062**	0.111	0.086	**0.009**	0.076	0.030	**0.025**
one leg jump	0.16	**0.062**	0.07	0.138	0.109	**0.067**	0.068	0.025	**0.024**
mean	0.30	0.19	**0.072**	0.106	0.097	**0.042**	0.070	0.039	**0.020**

Table 7.2 Full interpolation comparison on 10 D-FAUST sequences. The Hausdorff, Chamfer and varifold distance are computed against ground truth sequences.

and end point of our 10 test mini-sequences as the input for our interpolation problem. This allows us to compare the obtained results to the full mini-sequences, seen as a ground truth motion. In Figure 7.3, we compare the results of our method with the two mesh autoencoder methods and the ground truth for one mini-sequence. Our method is successful at recovering the latent codes that represent the endpoints and producing interpolations that remain in the space of human shapes. We further perform a quantitative comparison of the methods by measuring the distance to the ground-truth sequences at each break point with respect to the evaluation metrics given in Section 7.4.2; these results are displayed in Table 7.2. One can clearly observe that our method again outperforms the other methods both qualitatively and quantitatively.

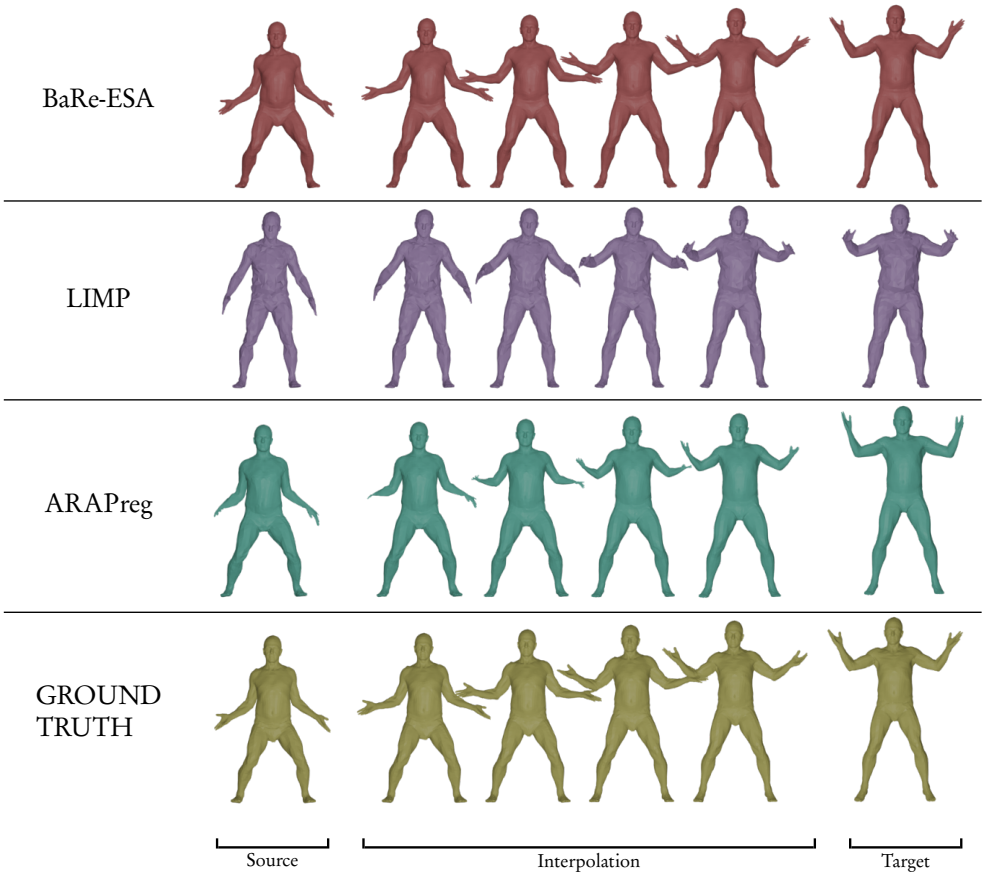


Figure 7.3 Interpolation results comparison between our method, LIMP, ARAPReg and the Ground Truth from D-FAUST. While the path produced by LIMP does not properly register the endpoints and the path produced by ARAPreg does not stay in the space of human bodies, BaRe-ESA successfully produces a path of human shapes whose endpoints match the source and target shapes.

7.4.5 Extrapolation

In the following, we consider the shape extrapolation problem, i.e., the task of predicting the future movement given a body shape and an initial movement (deformation). In our

	Hausdorff			Chamfer			Varifold		
	LIMP	ARAPReg	BaRe-ESA	LIMP	ARAPReg	BaRe-ESA	LIMP	ARAPReg	BaRe-ESA
punching	0.140	0.368	**0.102**	**0.030**	0.112	0.085	0.066	0.097	**0.025**
running on spot	0.287	0.309	**0.152**	0.11	0.177	**0.125**	0.071	0.079	**0.027**
running on spot b	0.264	0.415	**0.222**	0.138	0.179	**0.116**	0.071	0.083	**0.027**
shake arms	0.410	0.832	**0.273**	**0.080**	0.283	0.171	0.066	0.083	**0.027**
chicken wings	0.182	0.395	**0.092**	0.11	0.189	**0.092**	0.072	0.081	**0.018**
knees	1.13	0.282	**0.104**	0.23	0.205	0.081	0.321	0.072	**0.016**
knees b	0.489	0.319	**0.244**	0.17	0.215	**0.065**	0.066	0.071	**0.017**
jumping jacks	0.195	0.645	**0.091**	0.076	0.302	0.072	0.068	0.086	**0.027**
jumping jacks b	0.492	0.570	**0.122**	0.12	0.324	**0.061**	0.10	0.083	**0.029**
one leg jump	0.175	0.363	**0.167**	0.143	0.249	**0.136**	0.069	0.078	**0.025**
mean	0.391	0.452	**0.157**	0.118	0.214	**0.100**	0.103	0.082	**0.023**

Table 7.3 Full extrapolation comparison on 10 D-FAUST sequences. Again, the Hausdorff, Chamfer and varifold distance are computed against ground truth sequences.

Riemannian setting, this corresponds to solving the geodesic initial value problem using the method outlined in Section 7.2.1. From each of the 10 mini-sequences in our testing set, we recover the latent codes of the first two meshes in the sequence and use the first latent code and the difference of the codes are taken as input to our method. In Figure 7.4, we display the results of our extrapolation method, the extrapolations computed using LIMP and ARAPreg, and the original sequence (see the supplementary material for their corresponding animations). Our method is successful at producing extrapolations that capture the correct motion of the mesh without any extraneous motions that stay in space of human bodies. As with the interpolation comparison, we measure the distance to the ground-truth sequences at each breakpoint and display the results of the quantitative comparison in Table 7.3. Similar to the previous experiments, our method significantly outperforms the other methods.

7.4.6 Pose and Shape Disentanglement

By the construction of our basis, the first 130 dimensions of our latent space represent the body pose and the remaining 40 dimensions represent the body shape. Thus, our latent codes can be decomposed into coefficients that represent a change in body type and coefficients that represent a change in body pose. As a result, our framework is able to perform instant ***motion transfer***: once a motion is represented as a sequence of latent codes we simply replace the shape coefficients of each element of the sequence with the shape coefficients of the target shape. An example of this method in action is displayed in Figure 7.5.

7.4.7 Random Shape Generation

Another possible application of our framework is random shape generation. The idea is to use a data-driven distribution on the human shape tangent space. We first perform latent code retrieval on a subset of D-FAUST. We then compute, via finite differences, the initial tangent vector of each of these paths in the latent space, and separate these vectors into their pose and shape components. Onto each of these collections of tangent vectors, we fit a Gaussian mixture model, which is popular to generate human shapes (Bogo *et al.*, 2016; Omran *et al.*, 2018). We used 10 and 6 components for pose and shape respectively, which proved to be sufficient to get visually satisfying random shapes. The generation process consists in sampling a pose and shape vector in the tangent space and solving the corresponding geodesic initial value problem from

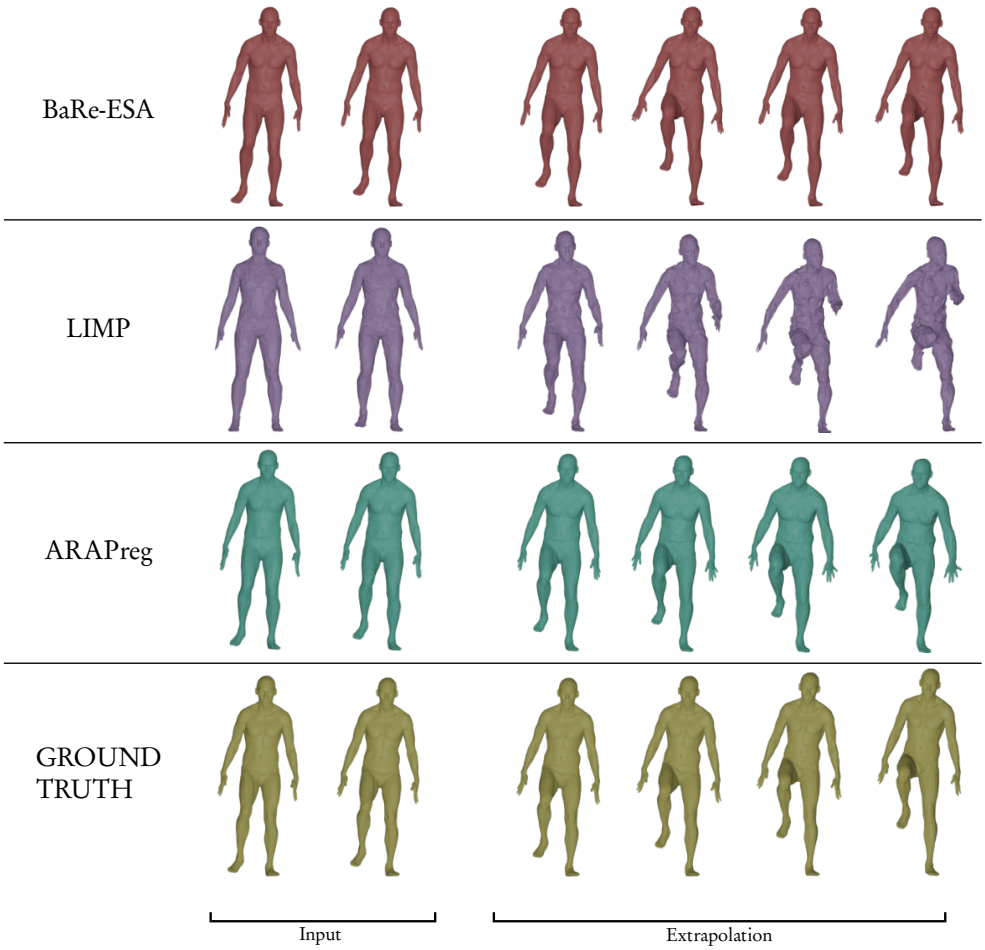


Figure 7.4 Extrapolation results comparison between our method, LIMP, ARAPReg and the Ground Truth from D-FAUST. While all three methods capture the primary motion of lifting a leg, the extrapolations produced by LIMP and ARAPreg include extraneous motions of the arms and slight changes in body type.

the template in the direction of the generated vector. We display a selection of 22 generated shapes in Figure 7.6.

7.5 Limitations

We point out, that the good results that we obtain in this Chapter and the last one come at the price of solving optimization problems to estimate interpolated or extrapolated geodesic paths for our metric. Another potential limitation of the proposed framework is the need for sufficiently rich training data: in the current implementation, we used the D-FAUST dataset for the generation of our motion and pose bases. In this dataset, there is, however, only very limited movement of the fingers/hands present and consequently, our motion basis is not able to faithfully represent such a movement. In future work, we plan to use additional datasets for the generation of the bases, which in turn will allow us to get a more complete representation

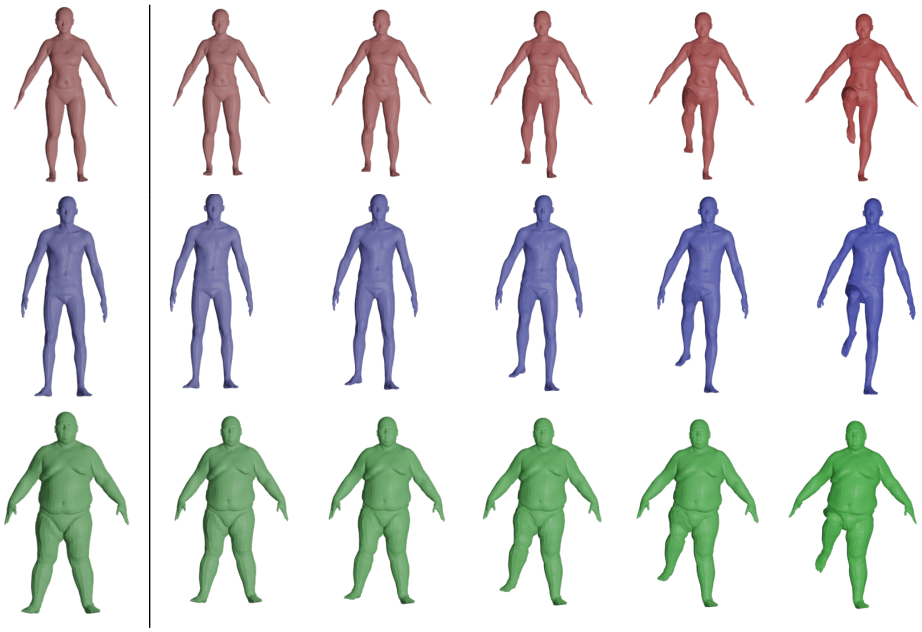


Figure 7.5 Motion Transfer: We display the original motion in the top row and the transfer of the motion to the target shapes in the second and third rows.

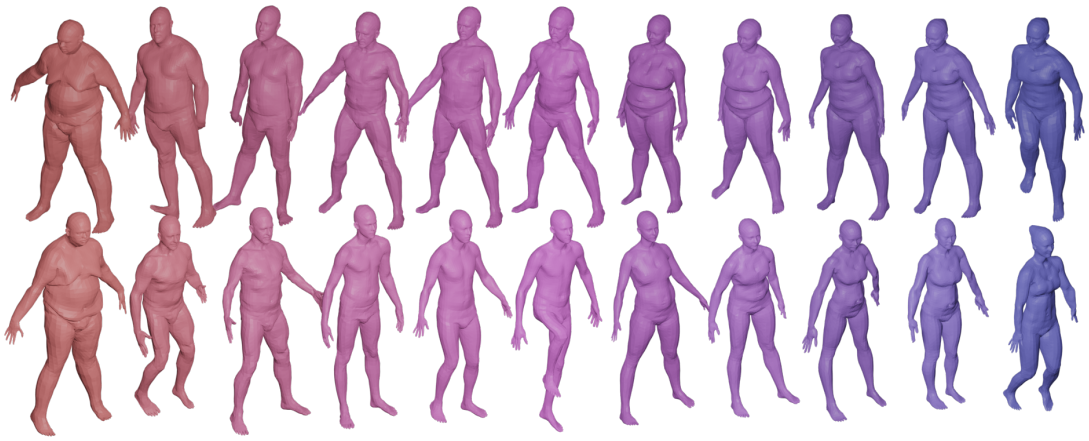


Figure 7.6 Random Shapes: 22 random shapes generated using a Gaussian mixture model on the space of initial velocities.

of all possible movements. Finally, we mention a simple yet potentially relevant extension of our model, namely to introduce two distinct Sobolev Riemannian metrics on the shape change and the pose change deformation fields. This comes with the idea of adapting the metric to the different nature of those deformations, and thus even better disentangling these quantities.

7.6 Conclusion

In this chapter, we proposed a novel framework for 3D human shape interpolation, extrapolation, and generation, that relies on learning deformation bases for body type and body pose changes combined with the use of a particular Riemannian structure on the latent space. Importantly, our method does not leave the shape space, does not require surfaces to have consistent meshes and vertex correspondences, and performs well even in the presence of imaging noise such as meshes with holes and/or missing parts. We showcased the different advantages of the proposed framework, in particular over recent deep learning approaches, for recovering and generating meaningful body shape trajectories.

Chapter 8

Conclusion and perspectives

8.1 Conclusion

In this thesis, we proposed new geometric approaches for human shape understanding, 3D shape sequence retrieval, and for the deformation of unparameterized human body scans. Our human shape representations take into account the specific geometry of human shapes and succeeded in the designated tasks. Firstly, we proposed a pose descriptor based on the convexity hypothesis that a human pose can be recognized from its convex hull. We then improved our results with a varifold-based human motion descriptor, which showed to be more robust to noise, than our first approach. Secondly, we proposed frameworks that not only were able to produce interpolation, and means of human shapes but also differentiate pose and shape deformations. The experiments showed that it outperformed deep learning concurring approaches. Finally, we extended our approach to unparameterized human body scans using varifold kernel metrics, showing outstanding results in even more applications, such as extrapolation or motion transfer. To conclude, we demonstrated that in the deep learning era, geometric approaches still have their place in human shape understanding. We are confident that our work can serve as a basis for some future approaches and obtain significant advances in human shape understanding.

8.2 Future works

In future works, we aim to investigate the following points:

8.2.1 Applications of our approaches

Video based 3D human body motion retrieval

Since our method in Chapter 4 shows good results in motion retrieval, we extend it to 2D video. We exploit the recent advancements in 2D-to-3D human body reconstruction. We reconstruct from the corresponding 3D human motion of a recorded video of me walking using a state-of-the-art single-view body reconstruction method, ICON (Xiu *et al.*, 2022). We then search for corresponding motions in the dataset using the method of Chapter 4. Figure 8.1 shows some examples of video and the corresponding 3D reconstructions using ICON.

Figure 8.1 Video and 3D human body reconstruction of ICON. We use the sequence of 3D shapes as a query.

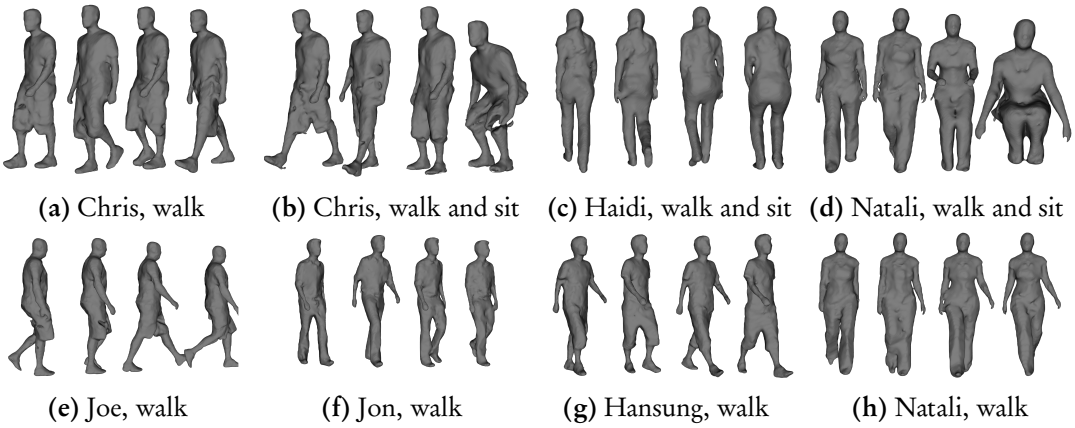


Figure 8.2 First tier of the query for Absolute Varifolds with the query being the reconstructed human shape motion shown in Figure 8.1. The results are ranked by closeness to the query using a given approach.

Figure 8.2 shows that half of the motions in the dataset are correctly retrieved (the wrong ones still contain a part of walking motions). These preliminary results show the robustness of the Gram-Hankel matrix of the 3D motion with respect to the noise due to the 3D reconstruction process. It demonstrates the ability of our method to exploit 3D information from only 2D video data. This opens the possibility of reducing material costs for the capture of 3D motion and bringing 3D shape analysis to new fields such as biomechanics and physiology.

Application to other surfaces

We believe it is possible to extend the ideas presented in this manuscript to other surfaces. It could be other articulated bodies like animals, objects that present non-isometric deformations like human faces and their expressions, or medical data such as cortical surfaces. Indeed, many components of our approaches (varifolds, Riemannian energies, ...) are not human-shape specific. Varifolds were applied successfully to cortical surfaces (Gori *et al.*, 2017) and their evolution (Bône *et al.*, 2019) in the medical context. On the other side, elastic metrics have been applied to human faces (Hartman *et al.*, 2022; Laga *et al.*, 2022).

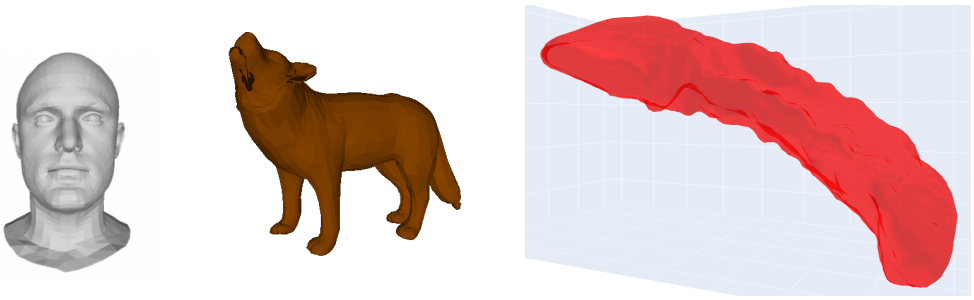


Figure 8.3 Other surfaces. On the left, is a human face. In the middle is a wolf surface. And on the right, a hippocampus surface.

8.2.2 Geometric deep learning

It is notable that our work is tied to a computational trade-off. If we want a framework that can be used for human deformation, we need to accept a loss in computational speed. This prevents us from being able to use frameworks of Chapters 6 and 7 in e.g. motion retrieval. Unlike our approach, geometric deep-learning auto-encoders do not have a computational trade-off. For example, the LIMP method (Cosmo *et al.*, 2020) generates interpolation of two shapes in a single feed-forward pass in the trained neural network. The retrieval of the latent code is also very fast and can be used for shape comparison. At this time, this comes at a significant cost in performance. In fact, in a recent work (Pierson *et al.*, 2022), we show that even for simple problems like the auto encoding of a toy dataset of ellipses, robust networks like PointNet can overlearn information about the surface parameterization, as illustrated in Figure 8.4.

Moreover, in the future, we will see the arrival of huge datasets of human motion. For example, the recently announced HuMMan dataset (Cai *et al.*, 2022) (expected to be released in 2023) will contain more than 400k 3D human body sequences executed by 1000 different subjects. The challenge of parameterization in geometric deep learning will have to be surmounted in order to profit from this quantity of data.

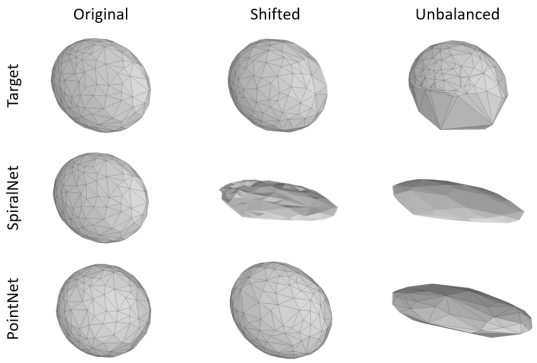


Figure 8.4 Task of learning latent spaces of ellipses via autoencoders. Ellipses are proposed as inputs of networks trained on a dataset with one fixed parameterization (original). Two types of connectivity preserving remeshings (shifted, unbalanced) are applied and the resulting meshes are given as inputs to the learned autoencoders, with Neural3DMM (called here SpiralNet) (Bouritsas *et al.*, 2019) and PointNet (Qi *et al.*, 2016) filters as a backbone. An ellipse with 3 different parameterizations is fed as input to the trained networks (first line). The reconstruction from the encoding and decoding processes of the network is displayed in line 2 for SpiralNet, and in last line for PointNet. We see that both networks fail in the case of an unbalanced remeshing, even PointNet yet its robustness against that kind of transformation.

Geometric deep learning frameworks could benefit from our approach to fulfill this objective. A few easy ideas can be declined from our manuscript and be incorporated into geometric deep learning approaches. First, we could use our human shape and motion descriptors as inputs of deep human body autoencoders. On the other hand, the Riemannian energies of paths defined in Chapters 6 and 7 could be used as loss functions to regularize deep learning latent space, following approaches that use other kinds of energies (Cosmo *et al.*, 2020; Huang *et al.*, 2021). Another possibility is to follow promising recent work parameterizing shape manifolds (Sassen *et al.*, 2023) to avoid solving the costly optimization problems encountered in Chapters 6 and 7.

8.2.3 A finer approach

A final limitation of our approach is that we always take the human shape as a global object, while we know that the human body is the concatenation of human body parts via its underlying skeleton and articulations. Moreover, human action is a superposition of multiple motions that happens at different scales.

This limits us from proposing an analysis of hand human motion from full human body data for example. This problem can be tackled from different angles. A natural way would be to design a natural segmentation of the human body from its underlying parts. For example, the recent SUPR body model (Osman *et al.*, 2022) considers body parts and proposes a human body model that unifies models for face, hand, foot and body. In a similar way, we could take profit of available datasets for each of the designated parts (Ranjan *et al.*, 2018b; Boyne *et al.*, 2022; Romero *et al.*, 2017) to extend our method to body parts.

Another, more abstract way is to adopt multi-scale or multi-resolution viewpoints. Using multi-resolution approaches is popular in computer graphics (Winkler *et al.*, 2010; Chu & Lee,

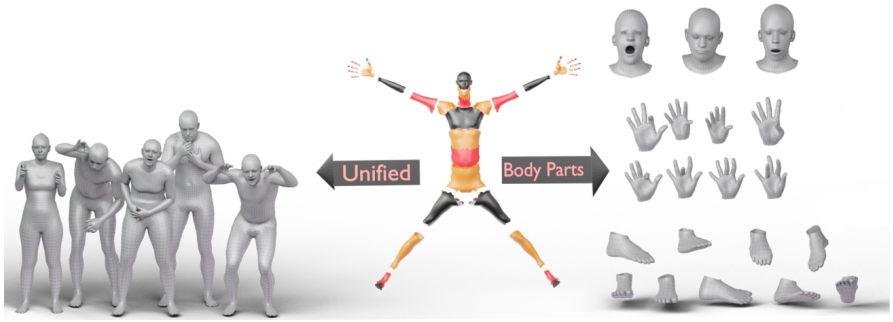


Figure 8.5 The human body is more than a shape, it is rather the unification of multiple body parts. Image of (Osman *et al.*, 2022).

2009) because it allows to both reduce the computational costs and take account of underlying geometries of shapes, using progressives downsamples and upsamples of the surface (Figure 8.6). This method is nowadays popular in geometric deep learning, in particular in mesh auto encoders (Hanocka *et al.*, 2019; Bouritsas *et al.*, 2019).

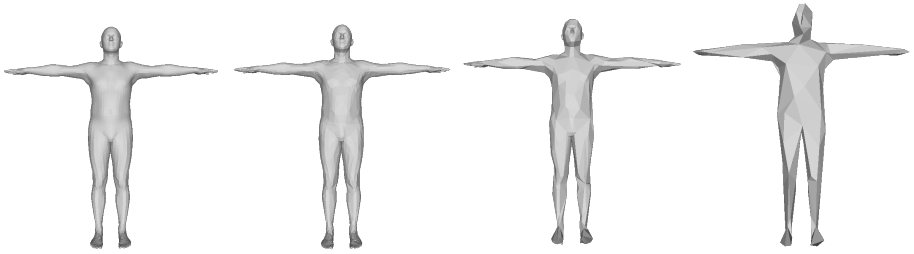


Figure 8.6 Progressive downsampling of a human body mesh. During the process, the structure of the human body is not lost, i.e. at last step, we can still guess the skeleton of the human body.

It seems possible that not only we can reduce the computational costs but also improve our algorithms by using such multi-scale approaches.

8.2.4 Theoretical guarantees

The geometric tools proposed by our approaches raise theoretical questions. For instance, in Chapter 4, we showed the valuable applications of the convexity hypothesis, as well as some of its limits. Determining a set of poses for which the hypothesis holds would open the possibility of reconstructing the pose from our extracted invariants. Moreover, Chapter 5 showed that the varifolds trajectories in the RKHS, corresponding to human body motions contain valuable information. It would be interesting to learn more about the properties of those trajectories, in particular for which sets of shapes we could use this information. Finally, the metrics presented in Chapter 6 and 7 opened the way to many applications, but there is still no geodesic completeness theorems for those metrics (interpolating between two surfaces could leave the surface space) as opposite to the case of curves (Bruveris, 2015). Finding the conditions for the choice of metric for geodesic completeness could help to understand better the failure cases of Riemannian approaches.

Bibliography

Achlioptas, Panos, Diamanti, Olga, Mitliagkas, Ioannis, & Guibas, Leonidas. 2018. Learning representations and generative models for 3d point clouds. *Pages 40–49 of: International conference on machine learning*. PMLR.

Alexa, Marc. 2003. Differential coordinates for local mesh morphing and deformation. *The Visual Computer*, **19**(2), 105–114.

Alexa, Marc, Cohen-Or, Daniel, & Levin, David. 2000. As-rigid-as-possible shape interpolation. *Pages 157–164 of: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*.

Alldieck, Thiemo, Xu, Hongyi, & Sminchisescu, Cristian. 2021. imghum: Implicit generative models of 3d human shape and articulated pose. *Pages 5461–5470 of: Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Allen, Brett, Curless, Brian, & Popović, Zoran. 2003. The space of human body shapes: reconstruction and parameterization from range scans. *ACM transactions on graphics (TOG)*, **22**(3), 587–594.

Anguelov, Dragomir, Srinivasan, Praveen, Koller, Daphne, Thrun, Sebastian, Rodgers, Jim, & Davis, James. 2005. SCAPE: Shape Completion and Animation of People. vol. 24. New York, NY, USA: Association for Computing Machinery.

Ankerst, Mihael, Kastenmüller, Gabi, Kriegel, Hans-Peter, & Seidl, Thomas. 1999. 3D Shape Histograms for Similarity Search and Classification in Spatial Databases. *Pages 207–226 of:* Güting, Ralf Hartmut, Papadias, Dimitris, & Lochovsky, Frederick H. (eds), *Advances in Spatial Databases, 6th International Symposium, SSD'99, Hong Kong, China, July 20-23, 1999, Proceedings*. Lecture Notes in Computer Science, vol. 1651. Springer.

Arnold, Vladimir. 2004. *Lectures on Partial Differential Equations*. Universitext. Berlin Heidelberg and Moscow: Springer-Verlag and PHASIS.

Aronszajn, Nachman. 1950. Theory of reproducing kernels. *Transactions of the American mathematical society*, **68**(3), 337–404.

Arun, K. S., Huang, T. S., & Blostein, S. D. 1987. Least-Squares Fitting of Two 3-D Point Sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **PAMI-9**(5), 698–700.

Aubry, Mathieu, Schlickewei, Ulrich, & Cremers, Daniel. 2011. The wave kernel signature: A quantum mechanical approach to shape analysis. *Pages 1626–1633 of: 2011 IEEE international conference on computer vision workshops (ICCV workshops)*. IEEE.

Aumentado-Armstrong, Tristan, Tsogkas, Stavros, Jepson, Allan, & Dickinson, Sven. 2019. Geometric Disentanglement for Generative Latent Shape Models. *Pages 8180–8189 of: 2019 IEEE/CVF International Conference on Computer Vision (ICCV)*.

Baek, Seung-Yeob, Lim, Jeonghun, & Lee, Kunwoo. 2015. Isometric shape interpolation. *Computers & Graphics*, **46**, 257–263.

Bansal, Sumukh, & Tatu, Aditya. 2018. Lie bodies based 3D shape morphing and interpolation. *Pages 1–10 of: Proceedings of the 15th ACM SIGGRAPH European Conference on Visual Media Production*.

Baran, Ilya, Vlasic, Daniel, Grinspun, Eitan, & Popović, Jovan. 2009. Semantic Deformation Transfer. *In: ACM SIGGRAPH 2009 Papers*. SIGGRAPH '09. New York, NY, USA: Association for Computing Machinery.

Barthelmé, Thomas. 2013. A natural Finsler-Laplace operator. *Israel Journal of Mathematics*, **196**(1), 375–412.

Basset, Jean, Boukhayma, Adnane, Wuhrer, Stefanie, Multon, Franck, & Boyer, Edmond. 2021. Neural Human Deformation Transfer. *Pages 545–554 of: International Conference on 3D Vision, 3DV 2021, London, United Kingdom, December 1-3, 2021*. IEEE.

Bauer, Martin, Harms, Philipp, & Michor, Peter W. 2011. Sobolev metrics on shape space of surfaces. *Journal of Geometric Mechanics*, **3**(4), 389.

Bauer, Martin, Harms, Philipp, & Michor, Peter W. 2012. Curvature weighted metrics on shape space of hypersurfaces in n-space. *Differential Geometry and its Applications*, **30**(1), 33–41.

Bauer, Martin, Charon, Nicolas, Harms, Philipp, & Hsieh, Hsi-Wei. 2021a. A numerical framework for elastic surface matching, comparison, and interpolation. *International Journal of Computer Vision*, **129**(8), 2425–2444.

Bauer, Martin, Charon, Nicolas, Harms, Philipp, & Hsieh, Hsi-Wei. 2021b. A Numerical Framework for Elastic Surface Matching, Comparison, and Interpolation. *Int. J. Comput. Vis.*, **129**(8), 2425–2444.

Bauer, Martin, Hartman, Emmanuel, & Klassen, Eric. 2022. The square root normal field distance and unbalanced optimal transport. *Applied Mathematics & Optimization*, **85**(3), 1–40.

Belongie, Serge, Malik, Jitendra, & Puzicha, Jan. 2000. Shape context: A new descriptor for shape matching and object recognition. *Advances in neural information processing systems*, **13**.

Ben Amor, Boulbaba, Su, Jingyong, & Srivastava, Anuj. 2016. Action recognition using rate-invariant analysis of skeletal shape trajectories. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **38**(1), 1–13.

Biasotti, S., Cerri, A., Bronstein, A., & Bronstein, M. 2016. Recent Trends, Applications, and Perspectives in 3D Shape Similarity Assessment. *Comput. Graph. Forum*, **35**(6), 87–119.

Blanz, Volker, & Vetter, Thomas. 1999. A morphable model for the synthesis of 3D faces. *Pages 187–194 of: Annual Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH)*.

Bogo, Federica, Romero, Javier, Loper, Matthew, & Black, Michael J. 2014. FAUST: Dataset and evaluation for 3D mesh registration. *Pages 3794–3801 of: Proceedings of the IEEE conference on computer vision and pattern recognition*.

Bogo, Federica, Kanazawa, Angjoo, Lassner, Christoph, Gehler, Peter, Romero, Javier, & Black, Michael J. 2016. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. *Pages 561–578 of: European conference on computer vision*. Springer.

Bogo, Federica, Romero, Javier, Pons-Moll, Gerard, & Black, Michael J. 2017. Dynamic FAUST: Registering human bodies in motion. *Pages 6233–6242 of: Proceedings of the IEEE conference on computer vision and pattern recognition*.

Bône, Alexandre, Martin, Benoît, Louis, Maxime, Colliot, Olivier, & Durrleman, Stanley. 2019. Hierarchical modeling of Alzheimer's disease progression from a large longitudinal MRI data set. *In: OHBM 2019-25th Annual Meeting of the Organization for Human Brain Mapping*.

Boothby, William Munger. 2003. *An introduction to differentiable manifolds and Riemannian geometry, Revised*. Vol. 120. Gulf Professional Publishing.

Boscaini, Davide, Masci, Jonathan, Melzi, Simone, Bronstein, Michael M, Castellani, Umberto, & Vandergheynst, Pierre. 2015. Learning class-specific descriptors for deformable shapes using localized spectral convolutional networks. *Pages 13–23 of: Computer graphics forum*, vol. 34. Wiley Online Library.

Bouritsas, Giorgos, Bokhnyak, Sergiy, Ploumpis, Stylianos, Bronstein, Michael, & Zafeiriou, Stefanos. 2019. Neural 3d morphable models: Spiral convolutional networks for 3d shape representation learning and generation. *Pages 7213–7222 of: Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Boyne, Oliver, Charles, James, & Cipolla, Roberto. 2022. FIND: An Unsupervised Implicit 3D Model of Articulated Human Feet. *In: British Machine Vision Conference (BMVC)*.

Brock, Andrew, Lim, Theodore, Ritchie, James Millar, & Weston, Nicholas J. 2016 (dec). Generative and Discriminative Voxel Modeling with Convolutional Neural Networks. *Pages 1–9 of: 3D Deep Learning Workshop; Advances in Neural Information Processing Systems*.

Bronstein, Alexander M, Bronstein, Michael M, Guibas, Leonidas J, & Ovsjanikov, Maks. 2011. Shape google: Geometric words and expressions for invariant shape retrieval. *ACM Transactions on Graphics (TOG)*, **30**(1), 1–20.

Bronstein, Michael M, & Kokkinos, Iasonas. 2010. Scale-invariant heat kernel signatures for non-rigid shape recognition. *Pages 1704–1711 of: 2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE.

Bronstein, Michael M., Bruna, Joan, LeCun, Yann, Szlam, Arthur, & Vandergheynst, Pierre. 2017. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, **34**(4), 18–42.

Bruveris, Martins. 2015. Completeness properties of Sobolev metrics on the space of curves. *Journal of Geometric Mechanics*, **7**(2), 125–150.

Buszard, Tim, Garofolini, Alessandro, Whiteside, David, Farrow, Damian, & Reid, Machar. 2020. Children's coordination of the "sweet spot" when striking a forehand is shaped by the equipment used. *Scientific reports*, **10**(1), 1–9.

Cai, Zhongang, Ren, Daxuan, Zeng, Ailing, Lin, Zhengyu, Yu, Tao, Wang, Wenjia, Fan, Xiangyu, Gao, Yang, Yu, Yifan, Pan, Liang, *et al.* 2022. HuMMan: Multi-Modal 4D Human Dataset for Versatile Sensing and Modeling. *arXiv preprint arXiv:2204.13686*.

Campbell, Lee W, & Bobick, Aaron F. 1995. Recognition of human body motion using phase space constraints. *Pages 624–630 of: Proceedings of IEEE international conference on computer vision*. IEEE.

Cao, Z., Hidalgo Martinez, G., Simon, T., Wei, S., & Sheikh, Y. A. 2019. OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Chadwick, John E, Haumann, David R, & Parent, Richard E. 1989. Layered construction for deformable animated characters. *ACM Siggraph Computer Graphics*, **23**(3), 243–252.

Charlier, Benjamin, Feydy, Jean, Glaunès, Joan Alexis, Collin, François-David, & Durif, Ghislain. 2021. Kernel Operations on the GPU, with Autodiff, without Memory Overflows. *Journal of Machine Learning Research*, **22**(74), 1–6.

Charon, Nicolas, & Trouvé, Alain. 2013. The varifold representation of nonoriented shapes for diffeomorphic registration. *SIAM journal on Imaging Sciences*, **6**(4), 2547–2580.

Cheeger, Jeff, Müller, Werner, & Schrader, Robert. 1984. On the curvature of piecewise flat spaces. *Communications in Mathematical Physics*, **92**(3), 405 – 454.

Chen, Xu, Zheng, Yufeng, Black, Michael J, Hilliges, Otmar, & Geiger, Andreas. 2021. SNARF: Differentiable forward skinning for animating non-rigid neural implicit shapes. *Pages 11594–11604 of: Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Chibane, Julian, Alldieck, Thiemo, & Pons-Moll, Gerard. 2020. Implicit functions in feature space for 3d shape reconstruction and completion. *Pages 6970–6981 of: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Choquet, Gustave. 1969. *Lectures on Analysis*. Vol. 3. Reading, Massachusetts: W.A. Benjamin, Inc.

Chu, Hung-Kuo, & Lee, Tong-Yee. 2009. Multiresolution mean shift clustering algorithm for shape interpolation. *IEEE transactions on visualization and computer graphics*, **15**(5), 853–866.

Chua, Chin Seng, & Jarvis, Ray. 1997. Point signatures: A new representation for 3d object recognition. *International Journal of Computer Vision*, **25**(1), 63–85.

Cosmo, Luca, Norelli, Antonio, Halimi, Oshri, Kimmel, Ron, & Rodolà, Emanuele. 2020. Limp: Learning latent shape representations with metric preservation priors. *Pages 19–35 of: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*. Springer.

Cox, Michael AA, & Cox, Trevor F. 2008. Multidimensional scaling. *Pages 315–347 of: Handbook of data visualization*. Springer.

Crane, Keenan. 2018. Discrete differential geometry: An applied introduction. *Notices of the AMS, Communication*, 1153–1159.

De Haan, Pim, Weiler, Maurice, Cohen, Taco, & Welling, Max. 2020. Gauge equivariant mesh CNNs: Anisotropic convolutions on geometric graphs. *arXiv preprint arXiv:2003.05425*.

Deng, Boyang, Lewis, John P, Jeruzalski, Timothy, Pons-Moll, Gerard, Hinton, Geoffrey, Norouzi, Mohammad, & Tagliasacchi, Andrea. 2020. Nasa neural articulated shape approximation. *Pages 612–628 of: European Conference on Computer Vision*. Springer.

DeWitt, Bryce S. 1967. Quantum Theory of Gravity. I. The Canonical Theory. *Phys. Rev.*, **160**(Aug), 1113–1148.

do Carmo, M. P. 1976. *An Introduction to Differential Geometry with Applications to Elasticity*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Ebin, D. G. 1970. The manifold of Riemannian metrics, In : Global Analysis, Berkeley, Calif., 1968. *Proc. Sympos. Pure Math.*, **15**, 11–40.

Eisenberger, M., Novotny, D., Kerchenbaum, G., Labatut, P., Neverova, N., Cremers, D., & Vedaldi, A. 2021. NeuroMorph: Unsupervised Shape Interpolation and Correspondence in One Go. *In: IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*.

Engel, Konrad, & Laasch, Bastian. 2020. Reconstruction of polytopes from the modulus of the Fourier transform with small wave length. *arXiv preprint arXiv:2011.06971*.

Fan, Haoqiang, Su, Hao, & Guibas, Leonidas J. 2017. A point set generation network for 3d object reconstruction from a single image. *Pages 605–613 of: Proceedings of the IEEE conference on computer vision and pattern recognition*.

Faugeras, Olivier D, & Hebert, Martial. 1986. The representation, recognition, and locating of 3-D objects. *The international journal of robotics research*, **5**(3), 27–52.

Fletcher, R. (Roger). 1987. *Practical methods of optimization*. Chichester ; New York : Wiley.

Freifeld, Oren, & Black, Michael J. 2012. Lie bodies: A manifold representation of 3D human shape. *Pages 1–14 of: European conference on computer vision*. Springer.

Fröhlich, Stefan, & Botsch, Mario. 2011. Example-driven deformations based on discrete shells. *Pages 2246–2257 of: Computer graphics forum*, vol. 30. Wiley Online Library.

Gadelha, M., Maji, S., & Wang, R. 2017. 3D Shape Induction from 2D Views of Multiple Objects. *Pages 402–411 of: 2017 International Conference on 3D Vision (3DV)*. Los Alamitos, CA, USA: IEEE Computer Society.

Gardner, R. J. 2006. *Geometric Tomography*. Cambridge University Pres.

Gavrila, Dariu M. 1999. The visual analysis of human movement: A survey. *Computer vision and image understanding*, **73**(1), 82–98.

Giachetti, Andrea, & Lovato, Christian. 2012. Radial symmetry detection and shape characterization with the multiscale area projection transform. *Pages 1669–1678 of: Computer graphics forum*, vol. 31. Wiley Online Library.

Gkalelis, Nikolaos, Kim, Hansung, Hilton, Adrian, Nikolaidis, Nikos, & Pitas, Ioannis. 2009. The i3dpost multi-view and 3d human action/interaction database. *Pages 159–168 of: 2009 Conference for Visual Media Production*. IEEE.

Gleyze, Valentin. 2019. Bruce Nauman: Contrapposto Studies. *Critique d'art. Actualité internationale de la littérature critique sur l'art contemporain*.

Goodfellow, Ian, Pouget-Abadie, Jean, Mirza, Mehdi, Xu, Bing, Warde-Farley, David, Ozair, Sherjil, Courville, Aaron, & Bengio, Yoshua. 2014. Generative Adversarial Nets. *In:* Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., & Weinberger, K.Q. (eds), *Advances in Neural Information Processing Systems*, vol. 27. Curran Associates, Inc.

Gori, Pietro, Colliot, Olivier, Marrakchi-Kacem, Linda, Worbe, Yulia, Poupon, Cyril, Hartmann, Andreas, Ayache, Nicholas, & Durrleman, Stanley. 2017. A Bayesian framework for joint morphometry of surface and curve meshes in multi-object complexes. *Medical image analysis*, **35**, 458–474.

Grinspun, Eitan, Hirani, Anil N, Desbrun, Mathieu, & Schröder, Peter. 2003. Discrete shells. *Pages 62–67 of: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*.

Groueix, Thibault, Fisher, Matthew, Kim, Vladimir G, Russell, Bryan C, & Aubry, Mathieu. 2018. 3d-coded: 3d correspondences by deep deformation. *Pages 230–246 of: Proceedings of the European Conference on Computer Vision (ECCV)*.

Hahner, Sara, & Garcke, Jochen. 2022. Mesh Convolutional Autoencoder for Semi-Regular Meshes of Different Sizes. *Pages 885–894 of: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*.

Hamza, A Ben, & Krim, Hamid. 2003. Geodesic object representation and recognition. *Pages 378–387 of: International conference on discrete geometry for computer imagery*. Springer.

Hanocka, Rana, Hertz, Amir, Fish, Noa, Giryes, Raja, Fleishman, Shachar, & Cohen-Or, Daniel. 2019. Meshcnn: a network with an edge. *ACM Transactions on Graphics (TOG)*, **38**(4), 1–12.

Hartman, Emmanuel, Sukurdeep, Yashil, Klassen, Eric, Charon, Nicolas, & Bauer, Martin. 2022. Elastic shape analysis of surfaces with second-order Sobolev metrics: a comprehensive numerical framework. *arXiv preprint arXiv:2204.04238*.

Hasler, Nils, Stoll, Carsten, Sunkel, Martin, Rosenhahn, Bodo, & Seidel, H-P. 2009. A statistical model of human pose and body shape. *Pages 337–346 of: Computer graphics forum*, vol. 28. Wiley Online Library.

Heeren, Behrend, Zhang, Chao, Rumpf, Martin, & Smith, William. 2018. Principal geodesic analysis in the space of discrete shells. *Pages 173–184 of: Computer Graphics Forum*, vol. 37. Wiley Online Library.

Ho, Jonathan, Jain, Ajay, & Abbeel, Pieter. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, **33**, 6840–6851.

Horn, Berthold Klaus Paul. 1984. Extended gaussian images. *Proceedings of the IEEE*, **72**(12), 1671–1686.

Huang, Peng, Hilton, Adrian, & Starck, Jonathan. 2010. Shape similarity for 3D video sequences of people. *International Journal of Computer Vision*, **89**(2-3), 362–381.

Huang, Qixing, Huang, Xiangru, Sun, Bo, Zhang, Zaiwei, Jiang, Junfeng, & Bajaj, Chandrajit. 2021. ARAPReg: An As-Rigid-As Possible Regularization Loss for Learning Deformable Shape Generators. *Pages 5815–5825 of: Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Jacobson, Alec, Panozzo, Daniele, *et al.* 2018. *libigl: A simple C++ geometry processing library*. https://libigl.github.io/.

Jermyn, Ian H., Kurtek, Sebastian, Klassen, Eric, & Srivastava, Anuj. 2012. Elastic Shape Matching of Parameterized Surfaces Using Square Root Normal Fields. *Pages 804–817 of: ECCV (5)*.

Jermyn, Ian H, Kurtek, Sebastian, Laga, Hamid, & Srivastava, Anuj. 2017. Elastic shape analysis of three-dimensional objects. *Synthesis Lectures on Computer Vision*, **12**(1), 1–185.

Johnson, A.E. 1997 (aug). *Spin-images: a representation for 3-D surface matching*. Ph.D. thesis, Carnegie Mellon University.

Kac, Mark. 1966. Can one hear the shape of a drum? *The american mathematical monthly*, **73**(4P2), 1–23.

Kacem, Anis, Daoudi, Mohamed, Amor, Boulbaba Ben, Berretti, Stefano, & Alvarez-Paiva, Juan Carlos. 2018. A novel geometric framework on gram matrix trajectories for human behavior understanding. *IEEE transactions on pattern analysis and machine intelligence*, **42**(1), 1–14.

Kaltenmark, Irene, Charlier, Benjamin, & Charon, Nicolas. 2017. A general framework for curve and surface comparison and registration with oriented varifolds. *Pages 3346–3355 of: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Kang, Sing Bing, & Ikeuchi, Katsushi. 1991. Determining 3-D object pose using the complex extended Gaussian image. *Pages 580–585 of: CVPR*, vol. 91.

Kaula, W. M. 1967. Theory of statistical analysis of data distributed over a sphere. *Reviews of Geophysics*, **5**(1), 83–107.

Kazhdan, Michael M., Funkhouser, Thomas A., & Rusinkiewicz, Szymon. 2003. Rotation Invariant Spherical Harmonic Representation of 3D Shape Descriptors. *Pages 156–164 of:* Kobbelt, Leif, Schröder, Peter, & Hoppe, Hugues (eds), *First Eurographics Symposium on Geometry Processing, Aachen, Germany, June 23-25, 2003*. ACM International Conference Proceeding Series, vol. 43. Eurographics Association.

Kilian, Martin, Mitra, Niloy J., & Pottmann, Helmut. 2007. Geometric Modeling in Shape Space. *Page 64–es of: ACM SIGGRAPH 2007 Papers*. SIGGRAPH '07. New York, NY, USA: Association for Computing Machinery.

Kingma, Diederik P., & Welling, Max. 2014. Auto-Encoding Variational Bayes. *In:* Bengio, Yoshua, & LeCun, Yann (eds), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Koenderink, Jan J, & Van Doorn, Andrea J. 1992. Surface shape and curvature scales. *Image and vision computing*, **10**(8), 557–564.

Kousholt, Astrid, & Schulte, Julia. 2021. Reconstruction of Convex Bodies from Moments. *Discrete & Computational Geometry*, **65**(1), 1–42.

Kurtek, Sebastian, Klassen, Eric, Gore, John C., Ding, Zhaohua, & Srivastava, Anuj. 2012. Elastic Geodesic Paths in Shape Space of Parameterized Surfaces. *IEEE Trans. Pattern Anal. Mach. Intell.*, **34**(9), 1717–1730.

Kurtek, Sebastian, Srivastava, Anuj, Klassen, Eric, & Laga, Hamid. 2013. Landmark-guided elastic shape analysis of spherically-parameterized surfaces. *Pages 429–438 of: Computer graphics forum*, vol. 32. Wiley Online Library.

Laga, Hamid, Xie, Qian, Jermyn, Ian H, & Srivastava, Anuj. 2017. Numerical inversion of SRNF maps for elastic shape analysis of genus-zero surfaces. *IEEE transactions on pattern analysis and machine intelligence*, **39**(12), 2451–2464.

Laga, Hamid, Padilla, Marcel, Jermyn, Ian H, Kurtek, Sebastian, Bennamoun, Mohammed, & Srivastava, Anuj. 2022. 4D Atlas: Statistical Analysis of the Spatio-Temporal Variability in Longitudinal 3D Shape Data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Lasseter, John. 1987. Principles of traditional animation applied to 3D computer animation. *Pages 35–44 of: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*.

Lemeunier, Clément, Denis, Florence, Lavoué, Guillaume, & Dupont, Florent. 2022. Representation learning of 3D meshes using an Autoencoder in the spectral domain. *Computers & Graphics*, **107**, 131–143.

Lévy, Bruno. 2008 (Feb.). *Géométrie Numérique*. Habilitation à diriger des recherches, Institut National Polytechnique de Lorraine - INPL.

Lewis, John P, Cordner, Matt, & Fong, Nickson. 2000. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. *Pages 165–172 of: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*.

Li, Binlong, Camps, Octavia I, & Sznaier, Mario. 2012. Cross-view activity recognition using hankelets. *Pages 1362–1369 of: 2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE.

Li, Xinju, & Guskov, Igor. 2005. Multiscale Features for Approximate Alignment of Point-based Surfaces. *Page 217 of: Symposium on geometry processing*, vol. 255. Citeseer.

Lian, Zhouhui, Godil, Afzal, Bustos, Benjamin, Daoudi, Mohamed, Hermans, Jeroen, Kawamura, Shun, Kurita, Yukinori, Lavoué, Guillaume, Nguyen, Hien Van, Ohbuchi, Ryutarou, Ohkita, Yuki, Ohishi, Yuya, Porikli, Fatih, Reuter, Martin, Sipiran, Ivan, Smeets, Dirk, Suetens, Paul, Tabia, Hedi, & Vandermeulen, Dirk. 2013. A comparison of methods for non-rigid 3D shape retrieval. *Pattern Recognit.*, **46**(1), 449–461.

Lipman, Yaron, Sorkine, Olga, Levin, David, & Cohen-Or, Daniel. 2005. Linear rotation-invariant coordinates for meshes. *ACM Transactions on Graphics (ToG)*, **24**(3), 479–487.

Livermore, P. W. 2012. The Spherical Harmonic Spectrum of a Function with Algebraic Singularities. *Journal of Fourier Analysis and Applications*, **18**(6), 1146–1166.

Lo Presti, Liliana, & La Cascia, Marco. 2015. Using Hankel matrices for dynamics-based facial emotion recognition and pain detection. *Pages 26–33 of: Proceedings of the IEEE conference on computer vision and pattern recognition workshops*.

Loper, Matthew, Mahmood, Naureen, & Black, Michael J. 2014. MoSh: Motion and shape capture from sparse markers. *ACM Transactions on Graphics (ToG)*, **33**(6), 1–13.

Loper, Matthew, Mahmood, Naureen, Romero, Javier, Pons-Moll, Gerard, & Black, Michael J. 2015. SMPL: A Skinned Multi-Person Linear Model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, **34**(6), 248:1–248:16.

Lowes, F. J. 1974. Spatial Power Spectrum of the Main Geomagnetic Field, and Extrapolation to the Core. *Geophysical Journal International*, **36**(3), 717–730.

Luo, Guoliang, Cordier, Frederic, & Seo, Hyewon. 2016. Spatio-temporal segmentation for the similarity measurement of deforming meshes. *The Visual Computer*, **32**(2), 243–256.

Mahmood, Naureen, Ghorbani, Nima, Troje, Nikolaus F, Pons-Moll, Gerard, & Black, Michael J. 2019. AMASS: Archive of motion capture as surface shapes. *Pages 5442–5451 of: Proceedings of the IEEE/CVF international conference on computer vision*.

Masci, Jonathan, Boscaini, Davide, Bronstein, Michael, & Vandergheynst, Pierre. 2015. Geodesic convolutional neural networks on riemannian manifolds. *Pages 37–45 of: Proceedings of the IEEE international conference on computer vision workshops*.

Mian, Ajmal S, Bennamoun, Mohammed, & Owens, Robyn. 2006. Three-dimensional model-based object recognition and segmentation in cluttered scenes. *IEEE transactions on pattern analysis and machine intelligence*, **28**(10), 1584–1601.

Mitchel, Thomas W, Kim, Vladimir G, & Kazhdan, Michael. 2021. Field convolutions for surface CNNs. *Pages 10001–10011 of: Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Monti, Federico, Boscaini, Davide, Masci, Jonathan, Rodola, Emanuele, Svoboda, Jan, & Bronstein, Michael M. 2017. Geometric deep learning on graphs and manifolds using mixture model cnns. *Pages 5115–5124 of: Proceedings of the IEEE conference on computer vision and pattern recognition*.

Muralikrishnan, Sanjeev, Chaudhuri, Siddhartha, Aigerman, Noam, Kim, Vladimir G., Fisher, Matthew, & Mitra, Niloy J. 2022. GLASS: Geometric Latent Augmentation for Shape Spaces. *In: Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Noguchi, Atsuhiro, Sun, Xiao, Lin, Stephen, & Harada, Tatsuya. 2021. Neural articulated radiance field. *Pages 5762–5772 of: Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Omran, Mohamed, Lassner, Christoph, Pons-Moll, Gerard, Gehler, Peter, & Schiele, Bernt. 2018. Neural body fitting: Unifying deep learning and model based human pose and shape estimation. *Pages 484–494 of: 2018 international conference on 3D vision (3DV)*. IEEE.

Osada, Robert, Funkhouser, Thomas A., Chazelle, Bernard, & Dobkin, David P. 2002. Shape distributions. *ACM Trans. Graph.*, **21**(4), 807–832.

Osman, Ahmed A A, Bolkart, Timo, & Black, Michael J. 2020. STAR: A Sparse Trained Articulated Human Body Regressor. *Pages 598–613 of: European Conference on Computer Vision (ECCV)*.

Osman, Ahmed A A, Bolkart, Timo, Tzionas, Dimitrios, & Black, Michael J. 2022. SUPR: A Sparse Unified Part-Based Human Body Model. *In: European Conference on Computer Vision (ECCV)*.

Otberdout, Naima, Ferrari, Claudio, Daoudi, Mohamed, Berretti, Stefano, & Del Bimbo, Alberto. 2022 (June). Sparse to Dense Dynamic 3D Facial Expression Generation. *Pages 20385–20394 of: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ovsjanikov, Maks, Ben-Chen, Mirela, Solomon, Justin, Butscher, Adrian, & Guibas, Leonidas. 2012. Functional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics (TOG)*, **31**(4), 1–11.

Papadakis, Panagiotis, Pratikakis, Ioannis, Theoharis, Theoharis, Passalis, Georgios, & Perantonis, Stavros. 2008. 3D Object Retrieval using an Efficient and Compact Hybrid Shape Descriptor. *In: Eurographics Workshop on 3D object retrieval*.

Paszke, Adam, Gross, Sam, Massa, Francisco, Lerer, Adam, Bradbury, James, Chanan, Gregory, Killeen, Trevor, Lin, Zeming, Gimelshein, Natalia, Antiga, Luca, Desmaison, Alban, Kopf, Andreas, Yang, Edward, DeVito, Zachary, Raison, Martin, Tejani, Alykhan, Chilamkurthy, Sasank, Steiner, Benoit, Fang, Lu, Bai, Junjie, & Chintala, Soumith. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Pages 8024–8035 of:* Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., & Garnett, R. (eds), *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc.

Pavlakos, Georgios, Choutas, Vasileios, Ghorbani, Nima, Bolkart, Timo, Osman, Ahmed A. A., Tzionas, Dimitrios, & Black, Michael J. 2019. Expressive Body Capture: 3D Hands, Face, and Body from a Single Image. *Pages 10975–10985 of: Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Pickup, D., Sun, X., Rosin, P. L., Martin, R. R., Cheng, Z., Lian, Z., Aono, M., Hamza, A. Ben, Bronstein, A., Bronstein, M., Bu, S., Castellani, U., Cheng, S., Garro, V., Giachetti, A., Godil, A., Isaia, L., Han, J., Johan, H., Lai, L., Li, B., Li, C., Li, H., Litman, R., Liu, X., Liu, Z., Lu, Y., Sun, L., Tam, G., Tatsuma, A., & Ye, J. 2016. Shape Retrieval of Non-rigid 3D Human Models. *International Journal of Computer Vision*, **120**(2), 169–193.

Pierson, Emery, Besnier, Thomas, Daoudi, Mohamed, & Arguillère, Sylvain. 2022. Parameterization Robustness of 3D Auto-Encoders. *In: Eurographics Workshop on 3D object retrieval*. The Eurographics Association.

Pishchulin, Leonid, Wuhrer, Stefanie, Helten, Thomas, Theobalt, Christian, & Schiele, Bernt. 2017. Building statistical shape spaces for 3d human modeling. *Pattern Recognition*, **67**, 276–286.

Plateau, Joseph. 1873. *Statique experimentale et theorique des liquides soumis aux seules forces moleculaires*. Gauthier-Villars.

Pons-Moll, Gerard, Romero, Javier, Mahmood, Naureen, & Black, Michael J. 2015. Dyna: A model of dynamic human shape in motion. *ACM Transactions on Graphics (TOG)*, **34**(4), 1–14.

Poonawala, A., Milanfar, P., & Gardner, R.J. 2002. A statistical analysis of shape reconstruction from areas of shadows. *Pages 916–920 of: Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers, 2002.*, vol. 1. Pacific Grove, CA, USA: IEEE.

Praun, Emil, & Hoppe, Hugues. 2003. Spherical parametrization and remeshing. *ACM transactions on graphics (TOG)*, **22**(3), 340–349.

Qi, Charles R, Su, Hao, Mo, Kaichun, & Guibas, Leonidas J. 2016. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. *arXiv preprint arXiv:1612.00593*.

Qi, Charles Ruizhongtai, Yi, Li, Su, Hao, & Guibas, Leonidas J. 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, **30**.

Qian, Guocheng, Li, Yuchen, Peng, Houwen, Mai, Jinjie, Hammoud, Hasan Abed Al Kader, Elhoseiny, Mohamed, & Ghanem, Bernard. 2022. *PointNeXt: Revisiting PointNet++ with Improved Training and Scaling Strategies*.

Ranjan, Anurag, Bolkart, Timo, Sanyal, Soubhik, & Black, Michael J. 2018a. Generating 3D faces using convolutional mesh autoencoders. *Pages 704–720 of: Proceedings of the European conference on computer vision (ECCV)*.

Ranjan, Anurag, Bolkart, Timo, Sanyal, Soubhik, & Black, Michael J. 2018b. Generating 3D faces using Convolutional Mesh Autoencoders. *Pages 725–741 of: European Conference on Computer Vision (ECCV)*.

Reuter, Martin, Wolter, Franz-Erich, & Peinecke, Niklas. 2006. Laplace-Beltrami spectra as 'Shape-DNA' of surfaces and solids. *Comput. Aided Des.*, **38**(4), 342–366.

Romero, Javier, Tzionas, Dimitrios, & Black, Michael J. 2017. Embodied Hands: Modeling and Capturing Hands and Bodies Together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, **36**(6).

Rumpf, Martin, & Wirth, Benedikt. 2013. Discrete geodesic calculus in shape space and applications in the space of viscous fluidic objects. *SIAM Journal on Imaging Sciences*, **6**(4), 2581–2602.

Rusu, Radu Bogdan, Blodow, Nico, & Beetz, Michael. 2009. Fast point feature histograms (FPFH) for 3D registration. *Pages 3212–3217 of: 2009 IEEE international conference on robotics and automation*. IEEE.

Rusu, Radu Bogdan, Bradski, Gary, Thibaux, Romain, & Hsu, John. 2010. Fast 3D recognition and pose using the Viewpoint Feature Histogram. *Pages 2155–2162 of: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*.

Saint, Alexandre, Kacem, Anis, Cherenkova, Kseniya, Papadopoulos, Konstantinos, Chibane, Julian, Pons-Moll, Gerard, Gusev, Gleb, Fofi, David, Aouada, Djamila, & Ottersten, Björn. 2020. Sharp 2020: The 1st shape recovery from partial textured 3D scans challenge results. *Pages 741–755 of: European Conference on Computer Vision*. Springer.

Santaló, Luis. 1976. *Integral geometry and geometric probability*. Encyclopedia of Mathematics and its Applications, vol. 1. Reading, Mass.-London-Amsterdam: Addison-Wesley Publishing Co.

Sassen, Josua, Heeren, Behrend, Hildebrandt, Klaus, & Rumpf, Martin. 2020. Geometric optimization using nonlinear rotation-invariant coordinates. *Computer Aided Geometric Design*, **77**, 101829.

Sassen, Josua, Hildebrandt, Klaus, Wirth, Benedikt, & Rumpf, Martin. 2023. Parametrizing Product Shape Manifolds by Composite Networks. *In: International Conference on Learning Representations*.

Scheepers, Ferdi, Parent, Richard E, Carlson, Wayne E, & May, Stephen F. 1997. Anatomy-based modeling of the human musculature. *Pages 163–172 of: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*.

Sellaroli, Giuseppe. 2017. An algorithm to reconstruct convex polyhedra from their face normals and areas. *arXiv preprint arXiv:1712.00825*.

Slama, Rim, Wannous, Hazem, & Daoudi, Mohamed. 2013. Extremal human curves: a new human body shape and pose descriptor. *Pages 1–6 of: 2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. IEEE.

Slama, Rim, Wannous, Hazem, & Daoudi, Mohamed. 2014. 3D human motion analysis framework for shape similarity and retrieval. *Image and Vision Computing*, **32**(2), 131–154.

Slama, Rim, Wannous, Hazem, Daoudi, Mohamed, & Srivastava, Anuj. 2015. Accurate 3D action recognition using learning on the Grassmann manifold. *Pattern Recognition*, **48**(2), 556–567.

Sorkine, Olga, & Alexa, Marc. 2007. As-rigid-as-possible surface modeling. *Pages 109–116 of: Symposium on Geometry processing*, vol. 4.

Sorkine, Olga, Cohen-Or, Daniel, Lipman, Yaron, Alexa, Marc, Rössl, Christian, & Seidel, H-P. 2004. Laplacian surface editing. *Pages 175–184 of: Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*.

Srivastava, Anuj, & Klassen, Eric P. 2016. *Functional and shape data analysis*. Vol. 1. Springer.

Starck, J., & Hilton, A. 2007. Surface Capture for Performance-Based Animation. *IEEE Computer Graphics and Applications*, **27**(3), 21–31.

Stein, Fridtjof, Medioni, Gérard, *et al.* 1992. Structural indexing: Efficient 3-D object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**(2), 125–145.

Su, Hang, Maji, Subhransu, Kalogerakis, Evangelos, & Learned-Miller, Erik. 2015. Multi-view convolutional neural networks for 3d shape recognition. *Pages 945–953 of: Proceedings of the IEEE international conference on computer vision*.

Su, Zhe, Bauer, Martin, Preston, Stephen C, Laga, Hamid, & Klassen, Eric. 2020a. Shape analysis of surfaces using general elastic metrics. *Journal of Mathematical Imaging and Vision*, **62**(8), 1087–1106.

Su, Zhe, Bauer, Martin, Klassen, Eric, & Gallivan, Kyle. 2020b. Simplifying transformations for a family of elastic metrics on the space of surfaces. *Pages 848–849 of: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*.

Sun, Jian, Ovsjanikov, Maks, & Guibas, Leonidas. 2009. A concise and provably informative multi-scale signature based on heat diffusion. *Pages 1383–1392 of: Computer graphics forum*, vol. 28. Wiley Online Library.

Terzopoulos, Demetri, Platt, John, Barr, Alan, & Fleischer, Kurt. 1987. Elastically deformable models. *Pages 205–214 of: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*.

Thomas, Hugues, Qi, Charles R., Deschaud, Jean-Emmanuel, Marcotegui, Beatriz, Goulette, François, & Guibas, Leonidas J. 2019. KPConv: Flexible and Deformable Convolution for Point Clouds. *Proceedings of the IEEE International Conference on Computer Vision*.

Tierny, Julien, Vandeborre, Jean-Philippe, & Daoudi, Mohamed. 2006. Invariant High Level Reeb Graphs of 3D Polygonal Meshes. *Pages 105–112 of: 3rd International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT 2006), 14-16 June 2006, Chapel Hill, North Carolina, USA*. IEEE Computer Society.

Tierny, Julien, Vandeborre, Jean-Philippe, & Daoudi, Mohamed. 2007. Reeb Chart Unfolding Based 3D Shape Signatures. *Pages 13–16 of:* Cignoni, Paolo, & Sochor, Jirí (eds), *28th Annual Conference of the European Association for Computer Graphics, Eurographics 2007 - Short Papers, Prague, Czech Republic, September 3-7, 2007*. Eurographics Association.

Tombari, Federico, Salti, Samuele, & Di Stefano, Luigi. 2010. Unique shape context for 3D data description. *Pages 57–62 of: Proceedings of the ACM workshop on 3D object retrieval*.

Tumpach, Alice Barbara, Drira, Hassen, Daoudi, Mohamed, & Srivastava, Anuj. 2016. Gauge Invariant Framework for Shape Analysis of Surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **38**(1), 46–59. arXiv: 1506.03065.

Turaga, Pavan, Veeraraghavan, Ashok, & Chellappa, Rama. 2008. Statistical analysis on Stiefel and Grassmann manifolds with applications in computer vision. *Pages 1–8 of: 2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE.

Turk, M.A., & Pentland, A.P. 1991a. Face recognition using eigenfaces. *Pages 586–591 of: Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.

Turk, Matthew A, & Pentland, Alex P. 1991b. Face recognition using eigenfaces. *Pages 586–587 of: Proceedings. 1991 IEEE computer society conference on computer vision and pattern recognition*. IEEE Computer Society.

van der Maaten, Laurens, & Hinton, Geoffrey. 2008. Visualizing Data using t-SNE. *JMLR*, **9**(86), 2579–2605.

Varol, Gül, Romero, Javier, Martin, Xavier, Mahmood, Naureen, Black, Michael J., Laptev, Ivan, & Schmid, Cordelia. 2017. Learning from Synthetic Humans. *Pages 4627–4635 of: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society.

Veinidis, Christos, Pratikakis, Ioannis, & Theoharis, Theoharis. 2017. On the retrieval of 3D mesh sequences of human actions. *Multimedia Tools and Applications*, **76**(2), 2059–2085.

Veinidis, Christos, Danelakis, Antonios, Pratikakis, Ioannis, & Theoharis, Theoharis. 2019. Effective descriptors for human action retrieval from 3D mesh sequences. *International Journal of Image and Graphics*, **19**(03), 1950018.

Verma, Nitika, Boyer, Edmond, & Verbeek, Jakob. 2018. Feastnet: Feature-steered graph convolutions for 3d shape analysis. *Pages 2598–2606 of: Proceedings of the IEEE conference on computer vision and pattern recognition*.

Virtanen, Pauli, Gommers, Ralf, Oliphant, Travis E., Haberland, Matt, Reddy, Tyler, Cournapeau, David, Burovski, Evgeni, Peterson, Pearu, Weckesser, Warren, Bright, Jonathan, van der Walt, Stéfan J., Brett, Matthew, Wilson, Joshua, Millman, K. Jarrod, Mayorov, Nikolay, Nelson, Andrew R. J., Jones, Eric, Kern, Robert, Larson, Eric, Carey, C J, Polat, İlhan, Feng, Yu, Moore, Eric W., VanderPlas, Jake, Laxalde, Denis, Perktold, Josef, Cimrman, Robert, Henriksen, Ian, Quintero, E. A., Harris, Charles R., Archibald, Anne M., Ribeiro, Antônio H., Pedregosa, Fabian, van Mulbregt, Paul, & SciPy 1.0 Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, **17**, 261–272.

Von-Tycowicz, Christoph, Schulz, Christian, Seidel, Hans-Peter, & Hildebrandt, Klaus. 2015. Real-Time Nonlinear Shape Interpolation. *ACM Trans. Graph.*, **34**(3).

Vranic, Dejan V. 2004. *3D model retrieval*. Ph.D. thesis, UNIVERSITÄT LEIPZIG INSTITUT FÜR INFORMATIK.

Weinland, Daniel, Ronfard, Remi, & Boyer, Edmond. 2006. Free viewpoint action recognition using motion history volumes. *Computer vision and image understanding*, **104**(2-3), 249–257.

Weyl, Hermann. 1946. *The classical groups*. Princeton, NJ: Princeton University Press.

Wieczorek, Mark A., & Meschede, Matthias. 2018. SHTools: Tools for Working with Spherical Harmonics. *Geochemistry, Geophysics, Geosystems*, **19**(8), 2574–2592.

Williams, Christopher, & Seeger, Matthias. 2001. Using the Nyström method to speed up kernel machines. *Pages 682–688 of: Proceedings of the 14th annual conference on neural information processing systems*.

Winkler, Tim, Drieseberg, Jens, Alexa, Marc, & Hormann, Kai. 2010. Multi-scale geometry interpolation. *Pages 309–318 of: Computer graphics forum*, vol. 29. Wiley Online Library.

Xiu, Yuliang, Yang, Jinlong, Tzionas, Dimitrios, & Black, Michael J. 2022 (June). ICON: Implicit Clothed humans Obtained from Normals. *Pages 13296–13306 of: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Xu, Dong, Zhang, Hongxin, Wang, Qing, & Bao, Hujun. 2005. Poisson shape interpolation. *Pages 267–274 of: Proceedings of the 2005 ACM symposium on Solid and physical modeling*.

Xu, Hongyi, Bazavan, Eduard Gabriel, Zanfir, Andrei, Freeman, William T, Sukthankar, Rahul, & Sminchisescu, Cristian. 2020. Ghum & ghuml: Generative 3d human shape and articulated pose models. *Pages 6184–6193 of: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Xu, Hongyi, Alldieck, Thiemo, & Sminchisescu, Cristian. 2021. H-nerf: Neural radiance fields for rendering and temporal reconstruction of humans in motion. *Advances in Neural Information Processing Systems*, **34**, 14955–14966.

Zhang, Xikang, Wang, Yin, Gou, Mengran, Sznaier, Mario, & Camps, Octavia. 2016. Efficient Temporal Sequence Comparison and Classification Using Gram Matrix Embeddings on a Riemannian Manifold. *Pages 4498–4507 of: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA: IEEE.

Zhang, Zhibang, Li, Guiqing, Lu, Huina, Ouyang, Yaobin, Yin, Mengxiao, & Xian, Chuhua. 2015. Fast as-isometric-as-possible shape interpolation. *Computers & Graphics*, **46**, 244–256.

Zhao, Hengshuang, Jiang, Li, Jia, Jiaya, Torr, Philip HS, & Koltun, Vladlen. 2021. Point transformer. *Pages 16259–16268 of: Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Zhou, Keyang, Bhatnagar, Bharat Lal, & Pons-Moll, Gerard. 2020a. Unsupervised Shape and Pose Disentanglement for 3D Meshes. *Pages 341–357 of:* Vedaldi, Andrea, Bischof, Horst, Brox, Thomas, & Frahm, Jan-Michael (eds), *Computer Vision – ECCV 2020*.

Zhou, Yi, Wu, Chenglei, Li, Zimo, Cao, Chen, Ye, Yuting, Saragih, Jason, Li, Hao, & Sheikh, Yaser. 2020b. Fully convolutional mesh autoencoder using efficient spatially varying kernels. *Advances in Neural Information Processing Systems*, **33**, 9251–9262.