Learning to generate shape from global-local spectra

Marco Pegoraro  
University of Verona  
marco.pegoraro@studenti.univr.it

Riccardo Marin  
Sapienza University of Rome  
marin@di.uniroma1.it

Umberto Castellani  
University of Verona  
umberto.castellani@univr.it

Simone Melzi  
Sapienza University of Rome  
melzi@di.uniroma1.it

Emanuele Rodolà  
Sapienza University of Rome  
rodola@di.uniroma1.it

Abstract

In this work, we present a new learning-based pipeline for the generation of 3D shapes. We build our method on top of recent advances on the so called shape-from-spectrum paradigm, which aims at recovering the full 3D geometric structure of an object only from the eigenvalues of its Laplacian operator. In designing our learning strategy, we consider the spectrum as a natural and ready to use representation to encode variability of the shapes. Therefore, we propose a simple decoder-only architecture that directly maps spectra to 3D embeddings; in particular, we combine information from global and local spectra, the latter being obtained from localized variants of the manifold Laplacian. This combination captures the relations between the full shape and its local parts, leading to more accurate generation of geometric details and an improved semantic control in shape synthesis and novel editing applications. Our results confirm the improvement of the proposed approach in comparison to existing and alternative methods.

1. Introduction

Controlling the generation of 3D shapes is a fascinating problem which unifies geometry and statistics at their finest. The interest is both theoretical and practical; we aim to understand better the rules lying “under the surface”, and, as a direct consequence, to solve real-world problems like automatic content generation, 3D shape reconstruction, body tracking [5, 11], among many others. This problem is as compelling as brave, and the research in the field has a long history. For several decades, many linear statistical approaches have been proposed [35, 23, 36], until deep learning methods took the stage unleashing powerful non-linear methods [5]. However, even if the results are getting more detailed every day, many underlying properties remain mysterious, hardly interpretable and controllable. Recently, a parallel arising trend is shedding new light on the study of geometric learning. Inverse spectral methods [8, 32] have shown that the eigenvalues of differential operators may uniquely represent a 3D object given a limited context (e.g., in the object class). One of the last advances in this field [26, 25] brings together for the first time the spectra with deep learning techniques. However, several points remain open and elicit our curiosity. First of all, it is not clear if an autoencoder architecture is needed as in [26, 25]. Nevertheless, what is indeed true, is that it blurs the relationship between the spectra and the generated models. Secondly, we believe that measuring the reconstruction quality just by using extrinsic reconstruction measures, misses the true nature of the link between 3D geometry and its spectrum. Thirdly, all the previous methods do not discuss how to include high-frequency information, and, lastly, all of them study a single differential operator.

In this work, we address all these questions by proposing a minimal, decoder-only architecture to generate shapes from their spectra directly. We show that this simple architecture produces convincing results and lets us better analyze the relationship between spectra and the associated 3D objects. Furthermore, we use this architecture to evaluate...
global and local spectra relations from both theoretical and practical perspectives. Our key observation is that local low frequencies encode global high frequencies, and can be used on characteristic regions to study semantics in our population. This is a crucial aspect of our work: we are not aiming to disentangle information, but to find relations between geometrical information.

We perform an extensive analysis on three different local operators and their combinations with the standard global Laplacian, concentrating our efforts (in contrast with previous works in the field) on two main tasks: direct shape-from-spectrum, and semantic control. An example of the latter is visualized in Fig. 1, where we combine the Laplacian spectrum of a male shape (in blue) with the eigenvalues of a local operator on the head of a female shape (in red), generating a new shape that globally reflects the semantic features of the blue shape (height and robustness) while possessing the local features of the red one (female). We remark that, being fully spectral, our method can deal with shapes that do not share the same connectivity. As a final point, we will show that this statistical correlation between local spectra and geometry is so strong that it holds even on unorganized point clouds, i.e., with noisy spectrum approximation and without known correspondence.

To summarize, our contribution is threefold:

1. We propose a decoder-only architecture that directly connects the spectrum to the 3D geometry. We do not only show that this simple approach outperforms previous more sophisticated methods, but also that this is more interpretable, giving us new theoretical insight on inverse spectral geometry problems;

2. For the first time, we address local semantic detail control in a shape-from-spectrum setup by proposing generation from multiple spectra. The combination of global and local information is also a novelty of our work, showing that this representation is capable of providing not only better reconstruction but also new application possibilities;

3. We propose a new dataset designed for analyzing inverse spectral methods, together with several extrinsic and intrinsic measures to give a novel point of view on the performance of this family of approaches.

All the data and the code will be made publicly available.

2. Related work

Generative models. 3D objects may have different representations depending on the application and, furthermore, each representation may carry large variety (e.g., vertex density). Researchers follow two main roads to learn from such data: simplify the data representation, or design models with invariance to such absence of structure. Among the endless list of proposed representations, we cite the most popular: voxels [41], triangle meshes [33, 39, 14], implicit functions [3, 4], and point clouds [31, 1]. However, each requires a specific architecture capable to take into account the domain characteristics. Even more complex is considering objects which undergo non-rigid deformations, and in particular the class of human bodies received great focus in the generative literature [43, 3, 15, 16]. Several of these methods are mainly focused on the shape (i.e., identity), rather than the pose; relying on standard animation techniques (such as linear blend skinning [20]) guarantees usability into standard computer graphics software. However, while previous works focus on reconstruction quality, few investigated the control and semantic generation of shapes. [2] proposes an autoencoder with a disentangled latent space, enabling a separate control of intrinsic and extrinsic deformations; [7] shows that plausibility of the generated shapes can be improved by promoting metric preservation in the loss function; [45] proposes an unsupervised technique to disentangle shape and pose in the latent space representation. Other related works address the generation of rigid composite-objects[22, 12, 42, 28] exploiting hierarchical neural network architectures to guide the control of shape-parts.

Shape from spectrum. Recently, the Laplacian eigenvalues have been used as a compact representation to recover and manipulate 3D geometry. According to a physical interpretation, the eigenfunctions of the Laplace operator on a surface relate to the evolution of waves over it, and the associated eigenvalues are the frequencies of such waves. These are determined uniquely by the intrinsic geometry of the shape, and are fully invariant to isometric deformations. However, the inverse problem (i.e., determining the intrinsic geometry from a set of Laplacian eigenvalues) has been an open question for a long time [17], with the negative result of Gordon and colleagues [13] posing a theoretical tombstone to the problem. The vision community has recently rediscovered interest in this problem from a practical perspective, with [8, 32] showing that this inverse problem can be solved through a complex optimization. The recent work [26], and its extension [25], replace the costly optimization with a data-driven framework, where a latent encoding is connected with the Laplacian spectrum via trainable maps. At test time, the network can instantaneously recover a shape from its spectrum. While we consider these works the closest to ours for its data-driven nature, [26, 25] limit their analysis to the standard Laplacian, without investigating localized operators [27, 6]. Importantly, the bandwidth used in [26, 25] (30 lowest eigenvalues) does not contain enough information on the shape geometry, and this information is spread by the network in an unclear way. Finally,
[26, 25] propose to solve the problem as a block of a complex and hard to interpret network, hiding the correlation between the input and the output.

**Our method.** Our method is straightforward as it relies on a simple decoder-only network, and considers a combination of different spectra. The only loss we use is a standard reconstruction loss. Without any other architectural component, our network discovers by itself the hidden pattern that links a spectrum to the 3D object. Hence, the statistical relations in our object class become prominent, providing better control of the generative process.

3. Background and notation

**Smooth setting.** In this setting, a 3D shape corresponds to a compact and connected Riemannian surface (2-dimensional manifold) \( \mathcal{X} \) embedded in \( \mathbb{R}^3 \) [10]. A ubiquitous tool in geometry processing is the Laplace-Beltrami Operator (LBO). Given a surface \( \mathcal{X} \), its LBO \( \Delta_{\mathcal{X}} \) is the generalization of the differential Laplacian operator to non-Euclidean domains. From now on, we will refer to this operator as Laplacian to streamline our notation. The Laplacian \( \Delta_{\mathcal{X}} \) is a positive semi-definite operator. From its eigendecomposition we obtain the eigenvalues \( \{\lambda_i\} \) with \( \lambda_i \in \mathbb{R} \), \( 0 \leq \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_{\infty} \), and associated orthogonal eigenfunctions \( \{\phi_i\} \). When \( \mathcal{X} \) is a manifold with boundary \( \partial \mathcal{X} \), we consider Dirichlet boundary conditions as done in [32]:

\[
\phi_i(x) = 0, \quad \forall x \in \partial \mathcal{X}.
\]

The ordered discrete sequence of Laplacian eigenvalues of \( \mathcal{X} \) is usually referred to as the spectrum of \( \mathcal{X} \). When \( \mathcal{X} \) is a 1D Euclidean domain, the eigenvalues coincide with the squares of frequencies of Fourier basis functions. Following this analogy, as a band-limited representation, only a truncated spectrum corresponding to the \( k \) eigenvalues with the smallest absolute values is considered. Usually, in geometry processing applications, \( k \) belongs to the interval \([30, 100]\). In this paper, we test different \( k \) in the range from 10 to 80. The Laplacian operator \( \Delta_{\mathcal{X}} \) is invariant to isometries, that is: \( \Delta_{\mathcal{X}} = \Delta_{(T,\mathcal{X})} \) for every isometry \( T \) (*i.e.* a deformation that preserves the metric). The spectrum inherits this invariance, since it is fully determined by \( \Delta_{\mathcal{X}} \).

In this work, we also consider a connected submanifold \( \mathcal{R} \subset \mathcal{X} \). \( \mathcal{R} \) is again a compact and connected Riemannian surface with boundary \( \partial \mathcal{R} \) (see inset figure for an illustration). \( \mathcal{R} \) inherits the metric from the complete manifold \( \mathcal{X} \) and could be similarly equipped with the Laplacian \( \Delta_{\mathcal{R}} \).

**Discrete setting.** We discretize smooth surfaces, submanifolds, and Laplacians in two alternatives ways: (i) **Triangle meshes:** \( \mathcal{M} = (X, F) \) with \( n \) vertices \( X \) and \( m \) triangular faces \( F \). As submanifold, we consider a subset of vertices with their local connectivity and preserve the faces. We adopt the cotangent formula to discretize the Laplacian [29].

(ii) **Unorganized point clouds:** \( P = (X) \). A submanifold is a subset of its points. We compute the Laplacian using the implementation of [38]

4. Method

A visualization of our pipeline is given in Fig. 2. This section outlines the proposed method providing details about its implementation and theoretical insights.

**Motivations.** As shown before, the spectrum of different operators (*e.g.*, the Laplacians) is compact, almost independent of the 3D representation, and stores informative geometric properties [34, 8, 32, 26, 25]. However, exploiting the Fourier analogy, considering only truncated subsets of the global Laplacian spectrum is limited by the low-pass representation. Including higher frequencies would be infeasible since they may dramatically differ even among similar objects. Instead, our intuition is to unify the global truncated spectrum with a *local low-band* spectrum. Our main insight is that restricting the computation of eigenvalues to a local region, even a low-band spectrum can reliably capture global high frequencies. Indeed, as depicted in Fig. 3, the aggregation of global and local spectra has a natural hierarchy given by the order of the eigenvalues of each operator and by the explicit subdivision in two subsets.

In our model, we avoid the use of an autoencoder (as done, instead, in [26, 25]) for two main reasons. First, we keep the neural architecture simple to save computational complexity. Secondly, we preserve the structural properties of our encoding, which is not trivial to obtain in the latent space of an autoencoder.

**Remark.** Our direct map from spectrum to 3D embedding emphasizes the statistical correlation between shape deformations and their impact on the spectra. In the next section, we show how crucial is this aspect to better control the generative expressiveness of the decoder, in comparison with the architecture proposed in [26, 25].

4.1. The proposed model

Given an input shape \( \mathcal{X} \), we aim to recover its 3D embedding and its geometric features directly from a combination of global and local spectra. The global truncated spectrum \( \Lambda_{\mathcal{X}} \) is computed from the Laplacian discretization. The local spectrum computation requires two steps: 1) retrieve an informative local region, coherently selected across all the shapes involved in training and test phases; 2) define a local operator and estimate its truncated spectrum.

\[ \text{https://github.com/nmwsharp/robust-laplacians-py} \]
tonian operator (PAT) the full shape (the Laplacian operator defined on the patch disjointed from the region $R \subset \mathcal{X}$). The model is mainly composed of a network $\Pi$ trained in a data-driven fashion (in red) that maps the spectral encoding to the 3d coordinates of the shapes.

Figure 2. The proposed model. We aim to recover a shape and its local characteristics directly from a combination of a global and local spectrum analytically computed from the input data (in gray). The model is mainly composed of a network $\Pi$ trained in a data-driven fashion (in red) that maps the spectral encoding to the 3d coordinates of the shapes.

Figure 3. Two examples of the computation of the proposed spectral encoding. From left to right: the shapes and the extracted sub-manifold depicted in red; the global and the local spectra respectively in blue and in red; the bar plot that represents the differences that compose our encoding.

Local region. We perform the first step by applying an automatic patch extractor to each shape defining a semantic sub-manifold $\mathcal{R} \subset \mathcal{X}$ specifically designed for each class of shapes (depicted on the left of Fig. 2). The desired region should be handy to retrieve on unseen data, and informative about the geometry and the details that characterize the objects of the class. For general objects, such as airplanes, some well-established segmentation approaches could solve this task both in a data-driven fashion [18, 37] or with more classical algorithms [9, 19]. This is possible also in more articulated cases like humans, identifying local regions like the head extractor proposed in [24].

Localized operator. Once we have retrieved the local region $\mathcal{R} \subset \mathcal{X}$, we localize an operator on this patch and compute its truncated spectrum $\Lambda_{\mathcal{R}}$. The most natural choice is the Laplacian operator defined on the patch disjointed from the full shape (PAT). We additionally consider the Hamiltonian operator (HAM) with a sharp potential localized in the submanifold [6], and the localized manifold harmonics (LMH) [27], a Hamiltonian-like operator whose eigenfunctions are orthogonal to the global Fourier basis.

Spectra encoding and aggregation. The values in the truncated global spectrum $\Lambda_\mathcal{X} \in \mathbb{R}^k$ are sorted non-decreasingly. We compute the differences:

$$d\lambda_\ell = \lambda_\ell - \lambda_{\ell-1}, \forall \ell \in \{2, \ldots, k\},$$

where $d\lambda_\ell \geq 0, \forall \ell$, and store them in a vector:

$$d\Lambda_\mathcal{X} = [d\lambda_2, \ldots, d\lambda_k] \in \mathbb{R}^{k-1}.$$  \hfill (2)

We do the same for $\Lambda_{\mathcal{R}} \in \mathbb{R}^h$, with $h \in \mathbb{N}^+$ obtaining $d\Lambda_{\mathcal{R}} \in \mathbb{R}^{h-1}$. Once we have the two vectors $d\Lambda_\mathcal{X}$ and $d\Lambda_{\mathcal{R}}$, we aggregate them generating our spectral encoding:

$$d\Lambda = [d\Lambda_\mathcal{X}; d\Lambda_{\mathcal{R}}] \in \mathbb{R}^{(k-1)+(h-1)}$$  \hfill (3)

A scheme of this computation is depicted in Fig. 3 and represented in green in the middle of Fig. 2. This encoding exploits the natural hierarchy owned by each set of eigenvalues. Moreover, differences between subsequent eigenvalues are more bounded with respect to their absolute values, which is a desirable property for the training.

Training phase. We aim to learn the decoder $\Pi$ which receives as input an aggregated spectrum $\Lambda$ and outputs the 3D embedding of the shape $\mathcal{X}$ that has $\Lambda$ as spectrum. Given a collection of training shapes, belonging to the same class, $\{\mathcal{X}_i\}_{i \in \mathcal{I}}$ for a certain list of indices $\mathcal{I}$, we can associate to each of them the spectral encoding $\{d\lambda_i\}_{i \in \mathcal{I}}$ following the process described above (depicted with gray color in Fig. 2). We implement the decoder $\Pi$ as a fully-connected network and train it to minimize a standard reconstruction loss:

$$Loss = \frac{1}{n} \sum_{i \in \mathcal{I}} \|\Pi(d\lambda_i) - X_i\|_F^2.$$  \hfill (5)

When we deal with point clouds, we substitute the Frobenius norm with the Chamfer distance defined in [1]. On the right of Fig. 2 we draw a high level scheme of this network (highlighted in red). We remark that while local regions are involved in the computation of the spectral encoding, we are not using any specific loss to guide the reconstruction of the corresponding local geometry.

Test phase. Once the model is trained, we can input to the network a previously unseen spectral encoding $d\Lambda$. The obtained 3D geometry reflect the one associated to the $\Lambda$ eigenvalues, but with the discretization of the training set.

Network architecture. The proposed network is composed of 4 fully connected layers. We report all the additional details about our implementation and training choices in the supplementary materials.
5. Results

5.1. Datasets

**Cube.** The CUBE dataset comprises 1000 meshes with 7350 vertices, which show 125 different local patterns and 8 scales, providing a controlled setup with orthogonal variations. Each sample was generated from a cube mesh combining a pattern variation and a scale variation. As submanifold $\mathcal{R}$ on each of these samples we select the face of the cube where we apply local variations.

**SURREAL.** To challenge our model on more realistic data, we collect 2337 human shapes from SURREAL [40]. As $\mathcal{R}$ on these human bodies we choose the head for three reasons: 1) it encodes the identity characteristics; 2) it is unique in the body and does not suffer from ambiguities and symmetries; 3) in contrast with cubes, we believe that head and body correlate. We are interested in verifying how this impacts our learning process.

**Airplanes.** The above datasets have a common discretization. To test if our model can discover the underlying pattern between spectra and deformation even in more general cases, we selected 448 airplanes from ShapeNet [44], we sampled 500 points from each one, producing unorganized point clouds for which the order of points does not correspond. As $\mathcal{R}$ we selected the tail segment.

We report all the additional details about the datasets we used in the supplementary materials.

5.2. Evaluation metrics

In our analysis, we consider two types of measures: **extrinsic** measures, which evaluate the Euclidean reconstruction error, and **intrinsic** measures, which are independent of the pose. These are computed between the ground truth target shapes and the ones generated by the model.

As extrinsic measures, we compute the same error optimized by the loss, referred to as $\text{MSE}$. With $\text{MSE-}$-$\mathcal{R}$ and $\text{MSE-}$-$\mathcal{R}^C$ we denote the same measure computed inside or outside $\mathcal{R}$. As intrinsic measures, we consider:

**Area:** the average difference of the area elements of each vertex, which relates to surface stretch.

**Metric:** the vertex-wise metric distortion, computed as the difference in the geodesic distances from a fixed set of 100 uniformly sampled points to all the points in the mesh.

**Align:** the MSE reconstruction error of the local region after the best rigid alignment, obtained by solving the Procrustes problem between the local patches.

The $\text{Area-}$-$\mathcal{R}$ and $\text{Metric-}$-$\mathcal{R}$ are the same as above, computed for the local region. We directly compare each method against the nearest-neighbor baseline; as done in [26], given the spectrum of a new test shape, we look in the training set for the spectrum which is the nearest in the L2 sense. Then, we consider as baseline output the training shape associated with this spectrum. To compare with this baseline we include $\text{ENN}$ which measures the MSE of the baseline, and $\text{EM} < \text{ENN}$ which indicates the percentage in which our results outperform the baseline.

5.3. Shape from spectrum

We test our method under different conditions to study in depth its generation from spectrum capability.

In Tables 1 and 2, for the Cube and the SURREAL dataset respectively, we collect the evaluation metrics (one per column) varying the local operator and the number of eigenvalues $k$ and $h \in \mathbb{N}$ (one per row). For the local region we propose three different operators: $\text{PAT } k+h$, $\text{HAM } k+h$, and $\text{LMH } k+h$, which concatenate $k$ LBO global eigenvalues to $h$ eigenvalues of a localized operator. In Figs. 4 and 5 we report qualitative comparisons respectively for the Cube and SURREAL datasets. Further, we summarize the main insights of our analysis:

**Global versus Local.** The standard eigenvalue representation of [26] is outperformed by mixed PAT encoding, regardless of different $k+h$ values. Nevertheless, even in the presence of orthogonal transformations, introducing a local representation helps the global reconstruction as well. We believe this is possible only if the network can relate each operator spectrum with a shape variation.

**Different localized operators.** We see that maximizing the locality of the considered representation is beneficial. PAT emerges as the best representation, tightly followed by HAM which is almost equivalent. LMH performs the worst. The discussion on this point is further detailed in Section 5.4.

**Locality proportion.** Our experiments suggest that a balanced mix of global and local information provides the best representation for the inverse spectral problem. Also, increasing the number of eigenvalues is more beneficial in a mixed setup than in the standard LBO setup. We think the information rapidly faints in subsequent eigenvalues, while new operators provide a clearer pattern for the network to harvest.

**Autoencoder vs decoder-only.** We compare our architecture (32M parameters) against the one proposed in [26] (9M parameters) also by empowering its decoder and, coherently, its encoder (namely [26]$_{\text{img}}$ in the table, 90M parameters). We trained [26] using the absolute value of eigenvalues, as proposed in the original paper. Results show not only that our decoder approach is better than the full architecture even in the LBO setting, but that [26] is not equally capable of combining local information with its latent space.

As a final remark, we emphasize that extrinsic metrics are not always reflected in the intrinsic ones. While the extrinsic measures directly test the network on the purpose of
### Table 1. Reconstruction error on the CUBE dataset. All the numbers should be multiplied by 1e-6. For each column, we highlighted the top three results in red with decreasing intensity.

| Method | MSE | MSE-\(R\) | MSE-\(R\) | ENN | EM < ENN | Area \((\times 10^4)\) | Metric \((\times 10^3)\) | Align | Area-\(R\) \((\times 10^4)\) | Metric-\(R\) \((\times 10^3)\) |
|--------|-----|-----------|-----------|-----|---------|----------------|----------------|-------|----------------|----------------|
| LBO 30 | 16.5 | 0.65 | 15 | 60% | 1.96 | 6.63 | 60.3 | 6.80 | 3.38 |
| LBO 50 | 10.7 | 62 | 15 | 65% | 1.80 | 6.41 | 60.2 | 6.76 | 3.33 |
| LBO 80 | 11.1 | 63.5 | 0.63 | 14 | 61% | 1.73 | 6.54 | 61.1 | 6.59 | 3.37 |
| PAT 20+10 | 6.65 | 37.8 | 0.42 | 309 | 88% | 1.78 | 4.81 | 37 | 6.25 | 2.42 |
| PAT 15+13 | 5.66 | 27.1 | 1.36 | 1090 | 95% | 2.31 | 3.96 | 25.5 | 6.01 | 1.75 |
| PAT 10+20 | 4.91 | 26.3 | 0.63 | 1720 | 100% | 1.43 | 3.59 | 24.7 | 5.37 | 1.79 |
| PAT 40+10 | 6.76 | 38.5 | 0.42 | 79 | 86% | 1.59 | 4.89 | 37.6 | 6.13 | 2.49 |
| PAT 25+25 | 3.59 | 19.1 | 0.5 | 2060 | 100% | 1.33 | 2.76 | 18.1 | 4.52 | 1.39 |
| PAT 10+40 | 6.70 | 23.4 | 3.36 | 1860 | 100% | 1.51 | 3.63 | 16.9 | 4.62 | 1.32 |

### Table 2. Reconstruction error on the SURREAL dataset. All the numbers should be multiplied by 1e-6. For each column, we highlighted the top three results in red with decreasing intensity.

| Method | MSE | MSE-\(R\) | MSE-\(R\) | ENN | EM < ENN | Area \((\times 10^4)\) | Metric \((\times 10^3)\) | Align | Area-\(R\) \((\times 10^4)\) | Metric-\(R\) \((\times 10^3)\) |
|--------|-----|-----------|-----------|-----|---------|----------------|----------------|-------|----------------|----------------|
| LBO 30 | 1.7 | 2.12 | 1.61 | 15.4 | 99.57% | 8.19 | 2.62 | 11 | 10.1 | 5.23 |
| PAT 25+5 | 1.1 | 1.42 | 1.03 | 19.6 | 100% | 15 | 2.21 | 4.8 | 23.9 | 3.32 |
| PAT 20+10 | 0.77 | 0.75 | 0.77 | 28.8 | 100% | 5.24 | 1.87 | 2.28 | 5.09 | 2.25 |
| PAT 15+13 | 0.71 | 0.5 | 0.76 | 39.1 | 100% | 4.28 | 1.81 | 2.42 | 4.94 | 2.27 |
| HAM 25+5 | 0.98 | 1.13 | 0.95 | 19 | 100% | 5.33 | 1.93 | 3.18 | 5.65 | 2.47 |
| HAM 20+10 | 0.73 | 0.67 | 0.75 | 27.4 | 100% | 5.11 | 1.86 | 2.43 | 5.09 | 2.29 |
| HAM 15+15 | 0.86 | 0.57 | 0.92 | 38.9 | 100% | 5.10 | 2.11 | 2.36 | 5.21 | 2.37 |
| LMH 25+5 | 2.7 | 2.49 | 2.75 | 29.9 | 100% | 10.6 | 3.37 | 14 | 11.7 | 5.71 |
| LMH 20+10 | 2.4 | 2.06 | 2.47 | 32.5 | 100% | 9.64 | 3.26 | 8.52 | 8.62 | 3.97 |
| LMH 15+15 | 1.5 | 0.98 | 1.61 | 31.2 | 100% | 7.15 | 2.02 | 3.96 | 6.16 | 2.67 |
| [26] PAT 15+15 | 3.1 | 3.89 | 2.92 | 51.3 | 100% | 19.81 | 4.46 | 19.37 | 32.93 | 8.44 |
| [26]\big PAT 15+15 | 2.51 | 1.8 | 2.67 | 51.3 | 100% | 16.9 | 4.32 | 19.20 | 22.25 | 9.28 |
| [26]\big LBO 30 | 2.51 | 2.26 | 2.56 | 15.09 | 99.96% | 15.74 | 4.35 | 15.26 | 22.87 | 7.24 |

5.4. Localized operators analysis

In Tables 1 and 2, we also report the performance of our method with different localized operators that differ in their support and dependence from the LBO. PAT is the only localized operator that we consider with local support and is independent of the LBO. The other two, HAM and LMH, have both global support and are computed localizing the LBO on a specific region. Moreover, LMH has an additional term of orthogonality that increases its dependency on the LBO. We claim that the different performances are due to their relationship with the LBO. PAT, being disjointed from the global geometry, can better express the local geometry behaviours giving the best results. On the contrary, HAM and LMH may experience some leaks of frequencies outside the local region. [32] has already shown that the HAM operator can approximate the LBO operator computed on a path with Dirichlet conditions. Our experiments confirm this relation: HAM gives only slightly worst results than PAT. Completely different is the case of LMH that strongly depends on the LBO spectrum. Its additional orthogonal condition may introduce information in the spectrum that our neural network considers redundant since it can be already extracted from the global spectrum. Even if the results in Tables 1 and 2 support our claim, a deeper analysis is required that we left as future direction.
5.5. Semantic Control by correlation discovery

We showed that introducing a local component in the representation helps reconstruct the local region as well as the entire shape. We believe this is due to three main facts: first, the network discovers a pattern between the local input representation and the local output geometry; second, the global part of the representation is less related to details and can correlate with the global 3D shape deformations; third, and most importantly: variations on the selected region occur together with nonlocal variations of other shape features; we learn this correlation. For example, several facial features of a human relate to body characteristics (e.g., male and female heads have different proportions, a wider face is more frequent in robust subjects, etc.). There is no limit in our fully-connected network to associate them to the local part of our input, if such a relation exists.

To stress this point we performed the experiment shown in Fig. 6. Given two shapes, $A$ and $B$, equipped with global spectra (rows) and local ones (columns), we recombine these representations into four combinations. Using our PAT15+15 network trained over SURREAL, in the top left and bottom right corners we reconstruct $A$ and $B$ from their original spectral encoding. In the other entries, we generate the new combined encodings: in the top right, we aggregate the local spectra of $B$ with the global one from $A$, while we do the opposite in the bottom left. While $A$ is a shape from SURREAL in T-pose, $B$ is from a different dataset (DYNA [30]) and also has a different pose. As can be seen, PAT15+15 performs well in the original reconstruction as well as in the head reconstruction even when swapped. However, we remark how the local and the global parts interact in their semantics: the head of $A$ glued on $B$ causes a thinner body (bottom-left) that is more robust than the original $A$. Instead, the head of $B$ on the body of $A$ generates a modification localized in the upper part of the body, to create a coherent body (top-right).

In Fig. 7, we perform a comparison with LBO 30 on
this task. We split the 30 given eigenvalues into 15 global frequencies and 15 local frequencies, and then we perform the same operations. The main difference is in the head reconstruction: PAT 15 + 15 creates the head according to the local spectrum, while LBO 30 follows the low frequency information (compare $A + A_R$ with $A + B_R$ in Fig. 7). Moreover, LBO 30 generates smoother shapes with less geometric details.

5.6. Applicability
In literature, spectral methods have been widely exploited. One of their main advantages is independence from the discrete representation. This property does not only solve the costly requirement of a point-to-point correspondence at interference time, but it also gives rise to high fidelity results from low-resolution shapes. We report a first example in Fig. 1 where we consider the spectra computed from shapes with a different connectivity encoded by two low resolution meshes. The output of our model is a mesh with a high resolution that respects the features of both the input shapes. In the supplementary material, we further stress this aspect of our approach showing more examples of reconstruction from shapes with different discrete representations (such as point clouds) and resolutions. Moreover, we show various examples of multi-spectral interpolation and interactive manipulation as done in [26, 25, 8].

5.7. Unorganized point clouds
During training, we request all the shapes to be in full point-to-point correspondence, which is of great help for the network to discover patterns in the data. To investigate if the relationship between the variation on the input eigenvalues and the variation of every point coordinate is strong enough to arise even without providing point correspondences, we remove the mesh connectivity and consider a noisy scenario of unordered point clouds of airplanes.

We considered the tail segment as the local region. The tails of airplanes are interesting because their size and shape vary, and identify specific categories (jet, passenger transport aircraft, etc.). Note that [26] requires redesigning the encoder to handle unsorted point clouds, while our model does not require any modification.

In Fig. 8, we propose a spectra swap similar to Fig. 6. For each of the first 15 global ($\lambda^g$) and local ($\lambda^l$) eigenvalues we consider the maximum and minimum values existing in our dataset. We denote these spectrum as $\Lambda_{\lambda^g}^{\max}$, $\Lambda_{\lambda^g}^{\min}$, $\Lambda_{\lambda^l}^{\max}$, $\Lambda_{\lambda^l}^{\min}$ respectively for the global and the local regions. We then compose 4 new spectra as the possible different combinations of the local and global parts. Modifying the global part (rows) changes the type of airplane (mainly the shape of wings and tail), but preserves the size of the tail (while its shape change to fit the different airplane). Touching the local part (columns), we obtain a smaller tail (due to the use of higher frequencies) but adherent to the type of airplane. Further examples with point clouds can be found in the supplementary materials.

6. Conclusions
This paper presented a novel approach to generate 3D shapes from a canonical and ubiquitous spectral representation. We highlighted several properties of local spectral operators and their relation with the standard Laplacian in this task. For the first time, a shape from spectrum pipeline has been performed from a mix of spectra of different operators. We have also shown the close relationship between spectra and the geometry from which they come, even in a noisy and unsupervised setup. We consider this theoretically exciting and helpful for shape manipulation especially in combining semantic characteristics of local and global parts. Dealing with multi-regions localization would be an interesting direction, considering parts with different semantic and proportions. While a complete analysis is beyond the scope of this paper, we include some preliminary analysis in the supplementary due to lack of space.

The main theoretical limitations of our study rely in not considering the spectra of extrinsic operators, like the Dirac operator [21]. This could be successfully injected into our pipeline, providing a mixture of intrinsic and extrinsic information. Also, we do not consider inter-class experiments since the spectrum may be ambiguous among different classes. We believe that our work may elicit discussion in the community on these topics. The main applicative limitations arise from the limitation of the intrinsic spectral representations in the presence of shapes with different
topology, significant noise or outliers. The research on these aspects is quite active in the community, and our method lends itself well to methodological progress in this respect. Finally, we consider our insight on the semantic shape composition as a promising starting point for future directions in shape synthesis.

References

[1] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas Guibas. Learning representations and generative models for 3d point clouds. In International Conference on Machine Learning, pages 40–49, 2018. 2, 4

[2] Tristan Aumentado-Armstrong, Stavros Tsogkas, Allan Jepson, and Sven Dickinson. Geometric disentanglement for generative latent shape models. In International Conference on Computer Vision (ICCV), 2019. 2

[3] Bharat Lal Bhattacharjee, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. Combining implicit function learning and parametric models for 3d human reconstruction. In European Conference on Computer Vision (ECCV), 2020. 2

[4] Ruojin Cai, Guandao Yang, Hadar Averbuch-Elor, Zekun Hao, Serge Belongie, Noah Snavely, and Bharath Haritharan. Learning gradient fields for shape generation. In European Conference on Computer Vision (ECCV), 2020. 2

[5] S. Chaudhuri, D. Ritchie, A. Shtern, A. Bronstein, and R. Kimmel. Hamiltonian operator for spectral shape analysis. IEEE Transactions on Visualization and Computer Graphics, 26(2):1230–1311, 2020. 2, 4

[6] Y. Choukroun, A. Shtern, A. Bronstein, and R. Kimmel. Hamiltonian operator for spectral shape analysis. IEEE Transactions on Visualization and Computer Graphics, 26(2):1230–1311, 2020. 2, 4

[7] Luca Cosmo, Antonio Norelli, Oshri Halimi, Ron Kimmel, and Emanuele Rodolà. LIMP: Learning Latent Shape Representations with Metric Preservation Priors. In European Conference on Computer Vision (ECCV), 2020. 2

[8] Luca Cosmo, Mikhail Panine, Arianna Rampini, Maks Ovsjanikov, Michael M Bronstein, and Emanuele Rodolà. Isospectralization, or how to hear shape, style, and correspondence. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 7529–7538, 2019. 1, 2, 3, 8

[9] M. Denitto, S. Melzi, M. Bicego, U. Castellani, A. Farinelli, M. A. T. Figueiredo, Y. Kleiman, and M. Ovsjanikov. Region-based correspondence between 3d shapes via spatially smooth biclustering. In 2017 IEEE International Conference on Computer Vision (ICCV), pages 4270–4279, 2017. 4

[10] Manfredo Perdigão do Carmo. Riemannian geometry. Birkhauser, Boston, 1992. 3

[11] Bernhard Egger, William AP Smith, Ayush Tewari, Stefanie Wohrer, Michael Zollhoefer, Thabo Beeler, Florian Bernard, Timo Bolkart, Adam Kortylewski, Sami Romdhani, et al. 3d morphable face models—past, present, and future. ACM Transactions on Graphics (TOG), 39(5):1–38, 2020. 1

[12] Matheus Gadelha, Giorgio Gori, Duygu Ceylan, Radomir Mech, Nathan Carr, Tamy Boubekeur, Rui Wang, and Subhransu Maji. Learning generative models of shape handles. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020. 2

[13] Carolyn Gordon, David L Webb, and Scott Wolpert. One cannot hear the shape of a drum. Bulletin of the American Mathematical Society, 27(1):134–138, 1992. 2

[14] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan Russell, and Mathieu Aubry. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2018. 2

[15] Boyi Jiang, Juyong Zhang, Jianfei Cai, and Jianmin Zheng. Disentangled human body embedding based on deep hierarchical neural network. 2020.

[16] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3D deformation model for tracking faces, hands, and bodies. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 8320–8329, 2018. 2

[17] Mark Kac. Can one hear the shape of a drum? The american mathematical monthly, 73(4P2):1–23, 1966. 2

[18] E. Kalogerakis, M. Averkiou, S. Maji, and S. Chaudhuri. 3d shape segmentation with projective convolutional networks. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 6630–6639. 4

[19] Yanir Kleiman and Maks Ovsjanikov. Robust structure-based shape correspondence. Comput. Graph. Forum, 38(1):7–20, 2019. 4

[20] John P Lewis, Matt Cordner, and Nickson Fong. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In Proceedings of the 27th annual conference on Computer graphics and interactive techniques, pages 165–172, 2000. 2

[21] Hsueh-Ti Derek Liu, Alec Jacobson, and Keenan Crane. A dirac operator for extrinsic shape analysis. Computer Graphics Forum, 36(5):139–149, 2017. 8

[22] Xinbing Liu, Zhizhong Liu, Yang Liu, and Matthias Zwicker. L2g auto-encoder: Understanding point clouds by local-to-global reconstruction with hierarchical self-attention. In Proceedings of the 27th ACM International Conference on Multimedia, 2019. 2

[23] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. ACM Trans. Graph., 34(6):248:1–248:16, 2015. 1, 11

[24] R. Marin, S. Melzi, E. Rodolà, and U. Castellani. Farm: Functional automatic registration method for 3d human bodies. Computer Graphics Forum, 39(1):160–173, 2020. 4

[25] Riccardo Marin, Arianna Rampini, Umberto Castellani, Emanuele Rodolà, Maks Ovsjanikov, and Simone Melzi. Instant recovery of shape from spectrum via latent space con-
nections. In *International Conference on 3D Vision (3DV)*, 2020. 1, 2, 3, 5, 6, 8, 11

[27] S. Melzi, E. Rodolà, U. Castellani, and M. M. Bronstein. Localized manifold harmonics for spectral shape analysis. *Computer Graphics Forum*, 37(6):20–34, 2018. 2, 4

[28] Kaichun Mo, Paul Guerrero, Li Yi, Hao Su, Peter Wonka, Niloy Mitra, and Leonidas Guibas. Structureenet: Hierarchical graph networks for 3d shape generation. *ACM Transactions on Graphics (TOG), Siggraph Asia 2019*, 38(6):Article 242, 2019. 2

[29] Ulrich Pinkall and Konrad Polthier. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics*, 2(1):15–36, 1993. 3

[30] Gerard Pons-Moll, Javier Romero, Naureen Mahmood, and Michael J Black. Dyna: A model of dynamic human shape in motion. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 34(4):120:1–120:14, Aug. 2015. 7

[31] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017. 2

[32] Arianna Rampini, Irene Tallini, Maks Ovsjanikov, Alex M Bronstein, and Emanuele Rodolà. Correspondence-free region localization for partial shape similarity via hamiltonian spectrum alignment. In *International Conference on 3D Vision (3DV)*, 2019. 1, 2, 3, 6

[33] Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J. Black. Generating 3D faces using convolutional mesh autoencoders. In *European Conference on Computer Vision (ECCV)*, 2018. 2

[34] Martin Reuter, Franz-Erich Wolter, and Niklas Peinecke. Laplace–beltrami spectra as ‘shape-dna’of surfaces and solids. *Computer-Aided Design*, 38(4):342–366, 2006. 3

[35] Kathleen M Robinette, Hans Daanen, and Eric Paquet. The caesar project: a 3-d surface anthropometry survey. In *Proc. Second International Conference on 3-D Digital Imaging and Modeling*, pages 380–386, Washington, DC, USA, oct 1999. IEEE. 1

[36] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6), Nov. 2017. 1

[37] Nicholas Sharp, Souhaib Attaiki, Keenan Crane, and Maks Ovsjanikov. Diffusion is all you need for learning on surfaces, 2020. 4

[38] Nicholas Sharp and Keenan Crane. A Laplacian for Non-manifold Triangle Meshes. *Computer Graphics Forum (SGP)*, 39(5), 2020. 3, 14

[39] Edgar Tretschk, Ayush Tewari, Michael Zollhöfer, Vladislav Golyanik, and Christian Theobalt. DEMEA: Deep Mesh Autoencoders for Non-Rigidly Deforming Objects. *European Conference on Computer Vision (ECCV)*, 2020. 2

[40] Gül Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *CVPR*, 2017. 5, 11

[41] Jiajun Wu, Chengkai Zhang, Tianfan Xue, William T Freeman, and Joshua B Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *Advances in Neural Information Processing Systems*, pages 82–90, 2016. 2

[42] Rundi Wu, Yixin Zhuang, Kai Xu, Hao Zhang, and Baoquan Chen. Pq-net: A generative part seq2seq network for 3d shapes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[43] Hongyi Xu, Eduard Gabriel Bazavan, Andrei Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. GHUM & GHUML: Generative 3D human shape and articulated pose models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6184–6193, 2020. 2

[44] Li Yi, Lin Shao, Manolis Savva, Haibin Huang, Yang Zhou, Qiwei Wang, Benjamin Graham, Martin Engelcke, Roman Klovak, Victor Lempitsky, et al. Large-scale 3d shape reconstruction and segmentation from shapenet core55. *arXiv preprint arXiv:1710.06104*, 2017. 5

[45] Keyang Zhou, Bharat Lal Bhatnagar, and Gerard Pons-Moll. Unsupervised shape and pose disentanglement for 3d meshes. In *European Conference on Computer Vision (ECCV)*, August 2020. 2
Supplementary Material

Abstract

As mentioned in the main manuscript, we report further details on our experiments and additional results in this document. We organize the supplementary material as follows: in Section 1 we report details about the data used in our experiments and architectures; in Section 2 we show results when the chosen local region has less semantic meaning (the forearm in the human bodies); in Section 3 we perform further experiments on the semantic control by local and global switching and interpolations; in Section 4 we show a static Figure of the interactive application shown in the attached video; in Section 5 we report more results on unorganized point clouds training.

1. Experimental Setup

1.1. Dataset

Here we report additional details about the datasets involved in our experiments.

**Cube.** In the CUBE dataset, the local variations are extrusions of simple geometric patterns (circle, ellipsis, square, and rectangle) applied on a selected face same for all cuboids. We vary dimensions and rotations of these patterns avoiding isometric shapes (that are indistinguishable by their eigenvalues). For the global variations, we scaled the cube along the dimension orthogonal to the face with local variation by a factor in the range $[0.6, 2]$, obtaining cuboids with different depths. In the inset figure, we visualize some of these examples.

**SURREAL.** The shapes in SURREAL are generated by SMPL [23], a standard generative template with 6890 vertices and two sets of parameters: one for the subject identity and one for its pose. Since pose changes generate near-isometric shapes, we set all the individuals in the same T-pose. The shape parameters are sampled from the ones available from SURREAL dataset [40].

**Airplanes.** In the Airplanes dataset we chose the segment of the tail as the local region because we think that the tail is a semantically significant region of the airplane: it is related to the airplane type (e.g., Boeing, Jet, Fighter) and its size.

1.2. Architecture and training details

Our architecture is a simple decoder composed of 4 fully connected layers. All the hidden layers use batch normalization followed by a selu activation, while the last layer has a linear activation. We report the number of nodes for each layer in Tab. 3. For the SURREAL dataset we add a dropout layer with a 0.1 drop rate to all hidden layers. We trained our network on 90% of the dataset and used the remaining 10% for testing. During training we used Adam optimizer with a learning rate of $2 \times 10^{-3}$ for the first 1000 epochs and then we reduce it to $1.8 \times 10^{-3}$ for the rest of the training. We fixed the maximum number of epoch in each dataset making sure each method reached convergence. In the second part of Table 3 we show the training parameters.

![Figure 9. Training loss comparison: our method in blue; in red the method proposed by [26].](image)

### Table 3. Networks parameters for the different datasets involved in our experiments.

|          | CUBE | SURREAL | Airplanes |
|----------|------|---------|-----------|
| Layer 1  | 258  | 258     | 258       |
| Layer 2  | 1024 | 512     | 1024      |
| Layer 3  | 2048 | 1536    | 2048      |
| Layer 4  | 22050| 20670   | 1500      |
| Output size | 7350 x 3 | 6890 x 3 | 500 x 3 |
| Number of epochs | 2000 | 1000    | 4500      |
| Batch size | 64   | 32      | 8         |

1.3. Comparison to the autoencoder of [26]

One of the main advantage of our method is the simplicity of the model: a single decoder composed of fully connected layers. This allows us to perform a more direct analysis of the linkage between spectral geometry processing and semantic modeling. On the contrary the model proposed by [26] is composed of an autoencoder enhanced with an invertible module blurring a similar analysis. In fact, the correspondence between the spectrum and the object geometry established by [26] passes through a latent space impacted by other components. Moreover, our specialized architecture performs better than the architecture proposed in [26] in synthesizing shapes from the spectrum. An other advantage of our model choice is the training. Fig. 9 shows the training loss curves of our model (in blue) and of [26] (in red). It emerges that our model not only reaches lower errors but has also a more stable training.
2. Does the selected region matters?

In the main manuscript, we defined a region for each class, claiming that they characterize the objects. In particular, we stated that this selection lets the network correlate between local geometric pattern and global features of the shape and that this relation has semantic meaning. As a counter-check of this, we report a result obtained by selecting an arbitrary region with few features, i.e., in which the details are less characterizing. In this experiment, we choose the right forearm as local region on the human body (visualized in the inset figure), and we trained the same model $PAT_{15+15}$ used on other SURREAL experiments. We refer to this method as $PAT_{F_{15+15}}$. In Fig. 10, we show the mean squared and the area error for an example of shape reconstruction comparing $PAT_{F_{15+15}}$ with $LBO_{30}$ and $PAT_{15+15}$. Concerning the $LBO_{30}$, we see that $PAT_{F_{15+15}}$ presents a similar intrinsic error on the head, while the error on the fore-arms is slightly improved (and also on other regions which may relate with the arm length as the hands and the feet). Extrinsically, the reconstruction error of $PAT_{F_{15+15}}$ is emphasized on the stomach. This area is well addressed by $LBO_{30}$, but also shows interesting correlation with the head region, as we show in Fig. 15. In all cases, $PAT_{15+15}$ performs significantly better. This experiment emphasizes the importance of the input representation and that some local regions contain more information on the whole shape than others.

2.1. Multiple regions

An other interesting direction to explore is the possibility to introduce more than one local spectra in our model. Our method naturally extends to this setting without the need of any modification, even if it introduces a further complexity (i.e., the relation between two local regions). In Fig. 11 we show a first results of reconstruction based on multiple regions: $PAT_{H_{10}+T_{10}+H_{10}'}$ considers both the head and the torso as regions and concatenate $k$ LBO global eigenvalues with $h$ local eigenvalues computed on the head and $h'$ local eigenvalues computed on the torso. The combination $10+10+10$ perform slightly worst w.r.t. $PAT_{15+15}$, while $15+15+15$ performs slightly better. The results show that the right balance on the involved eigenvalues is a component that needs further investigation and that we will inspect in future works.
Figure 10. Error comparison in shape reconstruction between LBO30, PAT15 + 15 and PAT15 + 15.

Figure 11. MSE comparison in shape reconstruction between LBO30, PAT15 + 15, PATH + T10 + 10 + 10 and PAT15 + 15.
3. Semantic Control

In Fig. 12 we report an additional result on semantic control. We start from a sparse point cloud (3445 vertices), depicted on the left, from which we compute the global spectrum with the robust Laplacian [38] and combine it with the local spectrum from a mesh representing a different subject in a different pose, visualized in the middle. On the right we show the resulting shape, which maintains the identity of the second one, but with a thinner body like the first shape. We remark that the network is trained only on meshes; thus we appreciate the robustness of our model also to unseen and noisy data.

3.1. Semantic interpolation

In this section, we show additional interpolation results for cubes and humans. Given a pair of shapes we perform 3 different interpolations: 1) on the whole spectral encoding $\Lambda$ that we propose; 2) on the global part of the representation $\Lambda_X$; 3) on the local part of the encoding referred to as $\Lambda_R$. We show a first example with the CUBE dataset in Fig. 13. We chose two shapes, visualized on the first row, with different depths and different local patterns. The interpolation of the whole encoding (second row) generates a smooth variation of both depth and pattern. When we change only the global part of the encoding (third row), this reflects only on the global scale by preserving the same pattern from the start shape. Finally, the local interpolation (last row) changes only the local pattern maintaining the same depth of the start shape. This experiment fully matches the underlying theory: the pattern and scale variation generated in the CUBE dataset are almost unrelated, and so their representations; the network learned the right association from the input representation without any hint and without mixing the two. We perform the same experiment on the SURREAL dataset in Fig. 14. We start from the mesh of a tall and thin male and interpolate it to the mesh of a shorter robust woman. The two shapes have different discretizations: the first shape has 2634 vertices, while the second has 1744 vertices. In the figure, we include a ruler in the rendering to emphasize the height variations. We observe that the $\Lambda$ interpolation (first row) changes the whole body. On the other hand, the $\Lambda_X$ interpolation (second row) changes only the height and the body structure without modification of the head. Finally, the local interpolation (third row) changes the head physiognomy and a portion of the body. This example shows that a head modification causes a change in the entire shape to keep a more consistent body structure. The main difference from the previous experiment on the cubes is the partial correlation between the local region and global body that our method naturally discovered.
Figure 13. Interpolation results for a pair of cubes. We also plot the spectral encoding associated with each shape as a bar plot. In the first row, we show the two input cubes. In the second row, the interpolation of the whole encoding. In the third row, the interpolation of only the global part (blue in the bar plots). In the last row, we only interpolate the local one (red in the bar plots). We highlight in green the part of the encoding that we are interpolating in each row and in red the values that we keep fix.

Figure 14. Different interpolations between two input human shapes (first row): Global+Local (A-second row), only Global(A_\Lambda-third row) and only Local(A_\mathcal{R}-fourth row). We add a ruler behind each shape to emphasise the height variation.
4. Interactive manipulation

Our method can be efficiently used in real-time to synthetize shapes and control their deformations by acting on the different parts of our representation. An example can be seen in the video attached to this document. In that short sequence, we have two sliders that modify the different components of the proposed spectral encoding. On the left side of the video, we show the spectrum that we input to the network, highlighting with brighter color the part subject to the current modification. In gray, we kept the original spectrum as a reference. On the right side of the video, we visualize the shape produced by our model. The color depicted on the surface encodes the difference between two subsequent modification frames; this visualization helps to identify where the modification represented by the sliders is acting on the 3D geometry. Moreover, in Fig. 15 we report an illustrative image of free manipulation provided by our model. We chose a reference shape and modified its global and local encoding separately. Similarly to the video, for each shape, we highlight on the surface the area variations encoded by the colors. In the first row, we decrease the global part of the encoding generating alterations scattered on the body, but with minimal interaction with the head. In the second row, we increase the values of the local part of our encoding obtaining a more feminine physiognomy and variations localized on the head and thorax, while the legs are almost left unchanged. This example shows similar behavior to Fig. 14 where modifications of the local part cause changes also on the body part correlated with the head, such as the stomach.
Figure 15. Free manipulation of the input. Given a shape (on the left), we decrease the global values ($\Lambda_X$-first row) and increase the local values ($\Lambda_R$-second row) separately. For each shape, we plot the area variations for each vertex and show the correspondent encoding as barplot (blue:global, red:local). We highlight in green the values that changes and in red the values that we keep constant.
Figure 16. A spectrum switch, similar to the interpolation shown in the main manuscript. On the left: the starting airplane and the corresponding output generated by our network. On the right, the second airplane. In the middle, from the top there are the reconstruction generated: using the whole second spectrum (\(\lambda\)); concatenating the global spectrum of the second with the local one of the first (\(\Lambda_X\)); concatenating the global spectrum of the first with the local spectrum of the second (\(\Lambda_R\)). Notice how the first two impact the whole plane, while the third changes the tail and preserves the global structure (e.g., wings and turbines).

Figure 17. The first (left) and the last (right) steps for the \(\Lambda_R\) interpolation depicted in Figure 19, with a close-up on the tails.

5. Unorganized pointclouds

Here we present other examples of our airplane experiments (Fig. 8 in the main paper).

In Fig. 18 we test our model by looking at the shape generated from the spectra obtained by interpolating the input spectral encoding of two shapes (depicted on the left). In the first row, we report the results from the interpolation of the whole spectral encoding. We can see that the deformation is smooth both in size (i.e., length of the structure) and features (i.e., turbines appearing, tail morphing). In the second row, we fix the local part of the encoding, and interpolate the global. Coherently, changing the whole structure also requires changing the tail structure (different kinds of airplanes have different tails). Finally, in the third row, we only manipulate the local part maintaining the global one. The local interpolation mainly impacts the tail region (a close-up is depicted in Fig. 17), which follows the interpolation pattern of other rows. Remarkably, other global aspects of the airplanes are only slightly modified (i.e., the turbines and the shapes of the wings are almost left unchanged). We consider this result significant, since the spectrum of the tail seems representative enough to relate with different airplanes. Moreover, our global plus local spectral encoding provides nice interpolation results.

In Fig. 16 we perform a spectrum switch. The two input planes have a similar tail but a different structure. Even in this subtle case, when we change only the local encoding, our method interpolates the two tails without modifying the airplane length, the presence of the turbines, and keeping the wings loyal to the starting plane. On the contrary, the global switch affects the whole plane like in the others interpolations experiments.

In Fig. 19 we report the same example of Fig. 18, but using a 20+10 network instead of a 15+15 one. The results are consistent, showing a certain resilience to different settings.
Figure 18. Casting different kinds of spectra interpolation into our network gives us different degrees of control. On the left, the models used as initial and final steps; on the right, we interpolated the entire spectral encoding ($\Lambda$-first row), only the global frequencies ($\Lambda_X$-second row), and only the local ones ($\Lambda_R$-third row).

Figure 19. Casting different kinds of spectra interpolation into our 20+10 network. On the left, the models used as initial and final steps; on the right, we interpolated the entire spectral encoding ($\Lambda$-first row), only the global frequencies ($\Lambda_X$-second row), and only the local ones ($\Lambda_R$-third row).