Supplementary materials
Universal Spectral Adversarial Attacks for Deformable Shapes

Arianna Rampini
Sapienza University of Rome
rampini@di.uniroma1.it

Franco Pestarini
Sapienza University of Rome
pestarini.1855627@studenti.uniroma1.it

Luca Cosmo
Sapienza University of Rome
cosmo@di.uniroma1.it

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

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

1. Architecture of the classifiers

Two types of networks were used in the tests, depending on the input. For meshes the implemented architecture is similar to the state-of-the-art encoder used by the CoMA autoencoder [4]. The structure of the network is shown in Table 1. It consists of 4 layers of ReLU-activated fast Chebyshev filters [1] with size $K = 6$, interleaved by mesh decimation via iterative edge collapse [2], and a final dense layer. We refer to this classifier simply as **ChebyNet**.

Layer	Input Size	Output Size
Convolution	3889×3	3889×128
Down-sampling	3889×128	1945×128
Convolution	1945×128	1945×128
Down-sampling	1945×128	973×128
Convolution	973×128	973×64
Down-sampling	973×64	487×64
Convolution	487×64	487×64
Fully Connected	31168	$\lvert\mathcal{C}\rvert$

Table 1: ChebyNet classifier architecture in detail for the SMAL dataset. $n = 3889$ is the number of vertices for the input meshes, and $|\mathcal{C}|$ is the number of classes.

For point clouds we used the **PointNet** classifier [3], composed by 4 layers of point convolution followed by batchnorm with ReLU, with layer output sizes $32 \to 128 \to 256 \to 512$. A maxpool operation is used to output a 512-dimensional vector, which is then reduced with a ReLU-activated fully connected network to dimensions: $512 \to 256 \to 128 \to 64 \to |\mathcal{C}|$, where $|\mathcal{C}|$ is the number of classes. Both ChebyNet and PointNet are trained to classify the subject identity for shapes in CoMA dataset [4], and the animal species for shapes in SMAL dataset [5]. The accuracy achieved in each case is reported in Table 2.

SMAL	ChebyNet	PointNet
train	100%	98.1%
test	100%	94.2%
remeshed	-	88.3%
CoMA	ChebyNet	PointNet
train	99.6%	99.0%
test	99.0%	99.2%

Table 2: Accuracy of the four considered classifiers in terms of fraction of correct predictions. For the SMAL dataset, PointNet was evaluated also on remeshed shapes from the test set, with a random number of vertices within 30% to 50% of the original ones.

2. Additional results

In Fig. 1 we show additional qualitative examples of universal attacks that due to lack of space were not included in the main manuscript.

Number of eigenvalues. We performed an analysis of the generalization capability of our method to previously unseen shapes at varying number of eigenvalues k. For each class we considered 15 shapes on which we performed the universal attack. We then transfer the deformation to 10 new shapes of the same class. Results on the **CoMA** dataset are reported in Table 3. Considering a larger number of eigenvalues k leads to an increase of success rate for the generalization. After $k = 60$, the performance decreases due to the difficulty to transfer the spectral deformation ρ, as measured by the *alignment error* $\epsilon_i = \|\sigma(X_i)(1+\rho) - \sigma(X_i + \Phi_i \alpha_i)\|$, where X_i is the original shape geometry and α_i are the perturbation coefficients. Since from the perturbed eigenvalues we synthesize novel shapes (the adversarial examples), we cannot compute a geometric error because a ground-truth 3D reconstruction does not exist. However, we can mea-

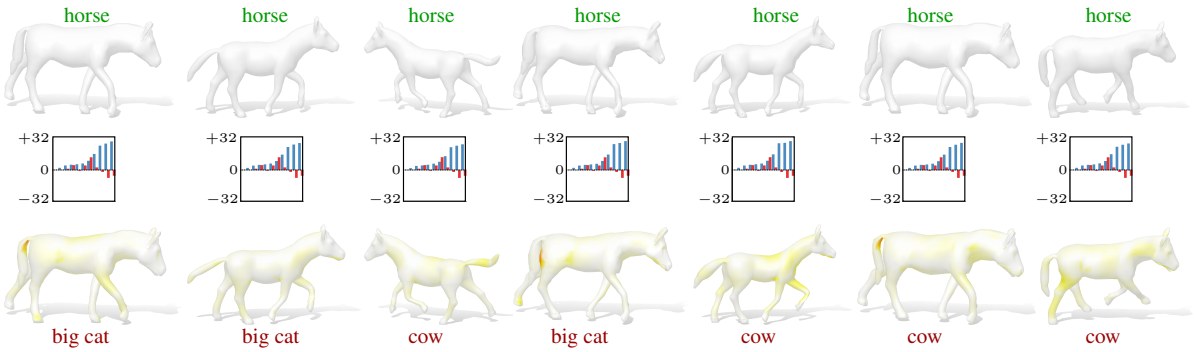

Figure 1: Example of universal adversarial attacks on PointNet over 7 shapes from the horse class of SMAL. The heatmap encodes curvature distortion, growing from white to dark red. Even if the original shapes are not isometric, as can be noted also from their spectra (blue bars), a universal spectral perturbation ρ (red bars, scaled by a factor 10^3) leads to misclassification.

k	success rate	alignment error
10	12%	2.65e-4
20	56%	1.33e-4
30	61%	1.96e-4
40	80%	2.72e-4
60	78%	3.06e-4
80	49%	5.84e-4
100	17%	7.01e-4

Table 3: Dependence of the generalization capability of our method on the number of used eigenvalues k. The *alignment error* is the absolute error between the target eigenvalues computed with ρ, and the eigenvalues of the deformed shapes; the *success rate* is the percentage of attacks that induce misclassification.

sure the alignment between the spectrum of the synthesized shapes and the target perturbed eigenvalues.

Point clouds. In Fig. 2 we show another example of generalization to point clouds; we compare the resulting deformation with the same perturbation applied to the corresponding mesh. To better appreciate the similarity between the two we exaggerated the perturbation of the universal attack by increasing the weight of the adversarial loss c.

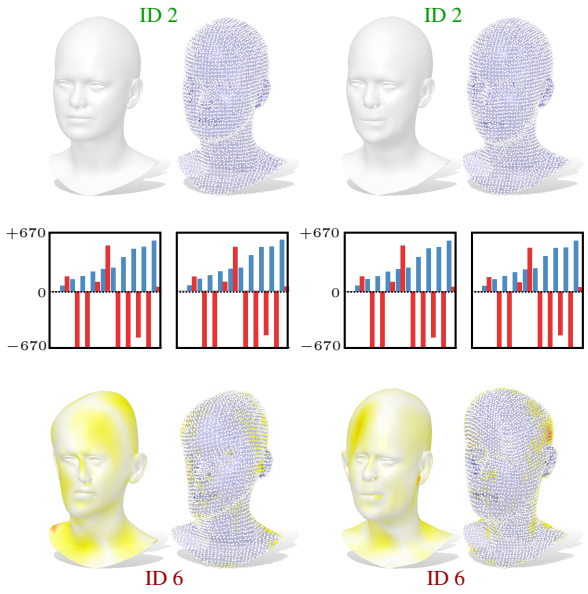

Figure 2: Examples of generalization to point clouds. The spectral perturbation ρ (red bars, scaled by a factor 10^3) was obtained on a set of 15 meshes (not shown). The deformation was then transferred to 2 unseen shapes discretized both as meshes (white on the left) and as point clouds (light blue on the right). The deformed shapes are shown in the last row. As we can see, for each shape the deformations induced by ρ are approximately the same regardless of the discretization. Note that here we intentionally enhanced the strength of the deformation (by increasing the weight of the adversarial loss) to better appreciate the similarity between the mesh and point cloud cases.

References

[1] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Proc. NIPS*, page 3844–3852, Red Hook, NY, USA, 2016. Curran Associates Inc. 1

[2] Michael Garland and Paul S Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 209–216, 1997. 1

[3] 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. 1

[4] 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. 1

[5] Silvia Zuffi, Angjoo Kanazawa, David Jacobs, and Michael J. Black. 3D menagerie: Modeling the 3D shape and pose of animals. In *Proc. CVPR*, July 2017. 1