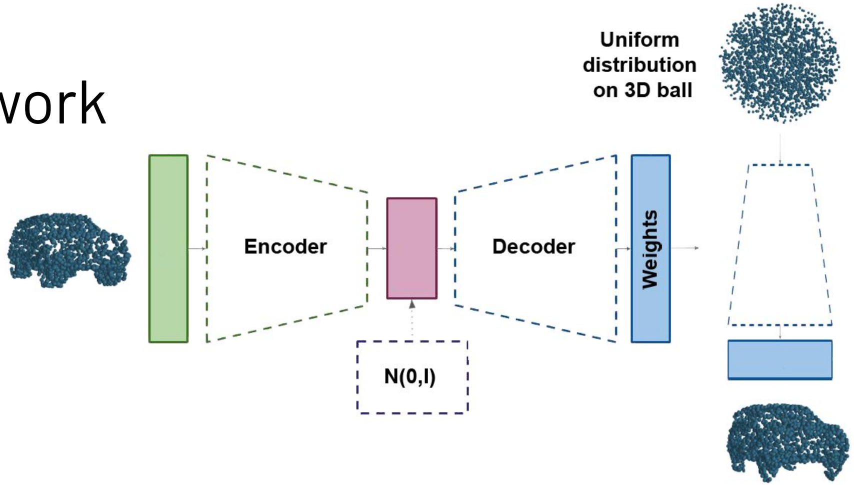
Hypernetwork approach to generating point clouds

Przemysław Spurek[1], Sebastian Winczowski[1], Jacek Tabor[1], Maciej Zamorski[2 4], Maciej Zięba[2 4], Tomasz Trzciński[3 4]

[1] Group of Machine Learning Research, Jagiellonian University
[2] Wrocław University of Science and Technology
[3] Warsaw University of Technology
[4] Tooploox

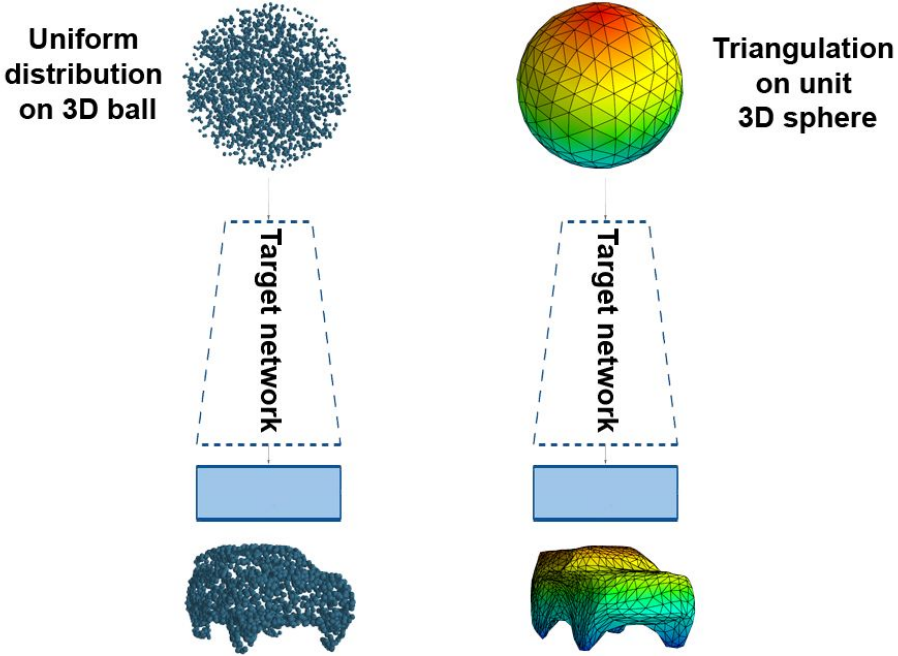
HyperCloud

- AAE architecture for hypernetwork
- PointNet as an Encoder
- Arbitrary prior on latent space
- Decoder produces weights for target network based on latent embedding
- Target network moves points from the uniform distribution on 3D ball to the 3D object



Easily extendible to meshes

- Use precomputed mesh instead of a point cloud
- Feeding vertices to the target network produces high-quality meshes
- No need for second mesh rendering



Experimental results

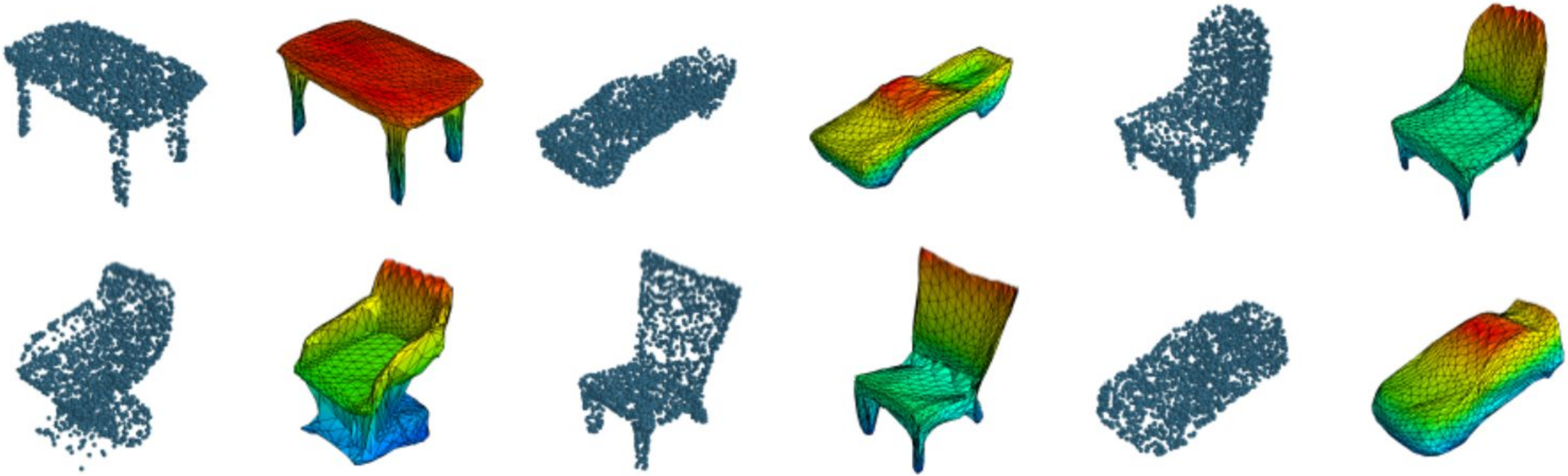


Figure: 3D point clouds and their mesh representations produced by HyperCloud

Experimental results

Table: Quality of representations by sampling from sphere (JSD ↓, MMD ↓, COV↑)

Class	Model	Sphere R	JSD	MMD		COV	
				CD	EMD	CD	EMD
Airplane	PointFlow	2.795	22.26	0.49	6.65	**44.69**	20.74
		3.136	26.46	0.60	6.89	39.50	19.01
		3.368	29.65	0.68	6.84	40.49	16.79
	HyperCloud	1.000	**9.51**	**0.45**	**5.29**	30.60	**28.88**
Chair	PointFlow	2.795	19.28	4.28	13.38	36.85	20.84
		3.136	22.52	4.89	14.47	32.47	17.22
		3.368	24.68	5.36	14.97	31.41	17.06
	HyperCloud	1.000	**4.32**	**2.81**	**9.32**	**40.33**	**40.63**
Car	PointFlow	2.795	16.59	1.60	8.00	20.17	17.04
		3.136	20.21	1.75	7.80	21.59	17.32
		3.368	24.10	1.96	8.35	18.75	17.04
	HyperCloud	1.000	**5.20**	**1.11**	**6.54**	**37.21**	**28.40**

Experimental results

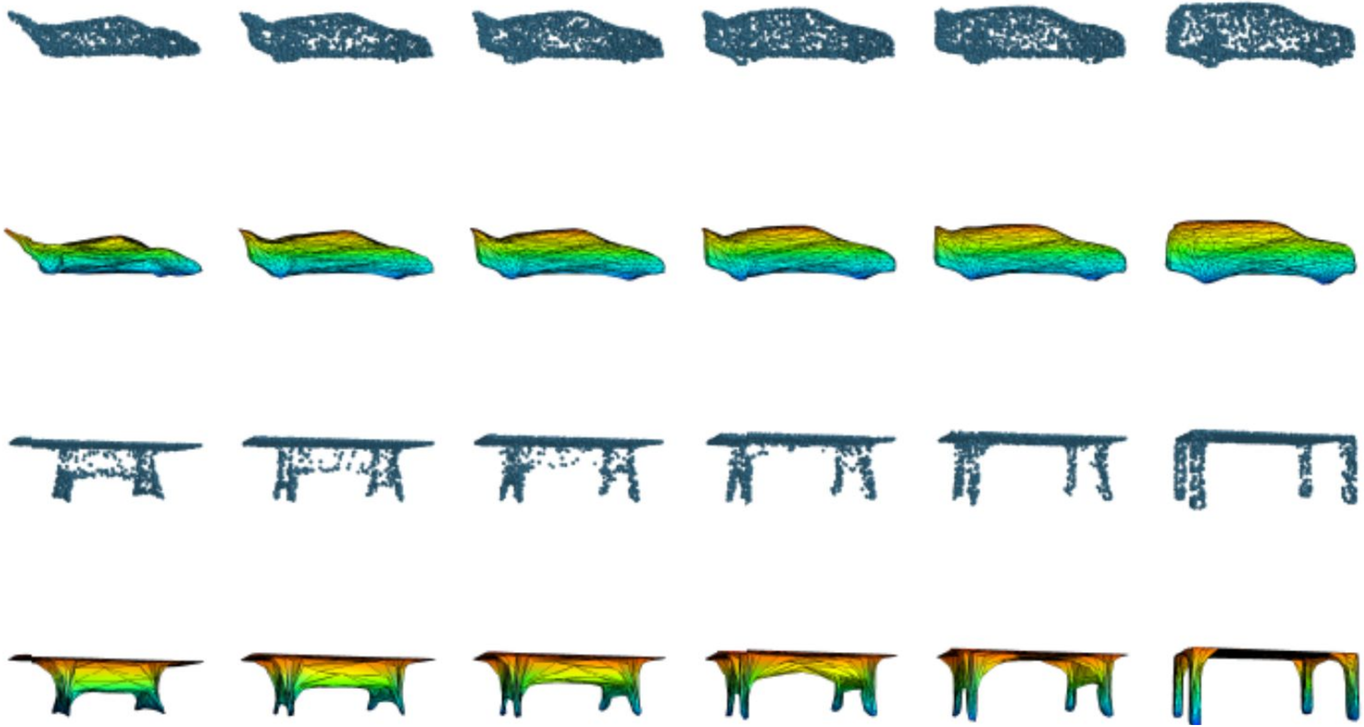


Figure: Interpolations between two 3D point clouds and their mesh representations

Conclusion

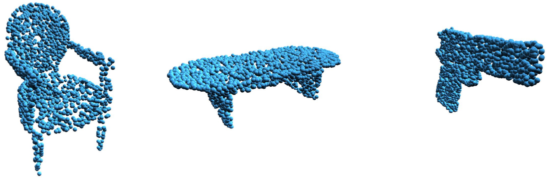
- We present a novel method for generating 3D point clouds
- Our approach is able to work not only on point clouds, but also on 3D meshes
- We leverage the hypernetworks to obtain simple architecture and fast end-to-end training
- Our model is able to generate shapes consisting of an arbitrary number of points or vertices

Point Clouds

- Objects represented as sets of real-valued points

$$= \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ \vdots & \vdots & \vdots \\ x_K & y_K & z_K \end{bmatrix} \in \mathbb{R}^{K \times 3}$$

- Unstructured
- Unordered - K! possible arrangements of K points
- One of the most common dataset is ShapeNet
  - 57k samples, 55 classes

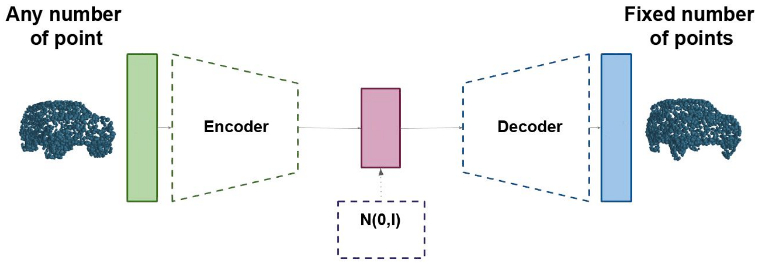


Related work: Hypernetworks

- Is training all parameters in (very) deep neural networks necessary?
- Instead train a smaller NN (hypernetwork) that generates weights for the (target) network
  - HyperNetworks (Ha et al. ICLR 2016)
  - Hypernetwork functional image representation (Klocek et al. ICANN 2019)

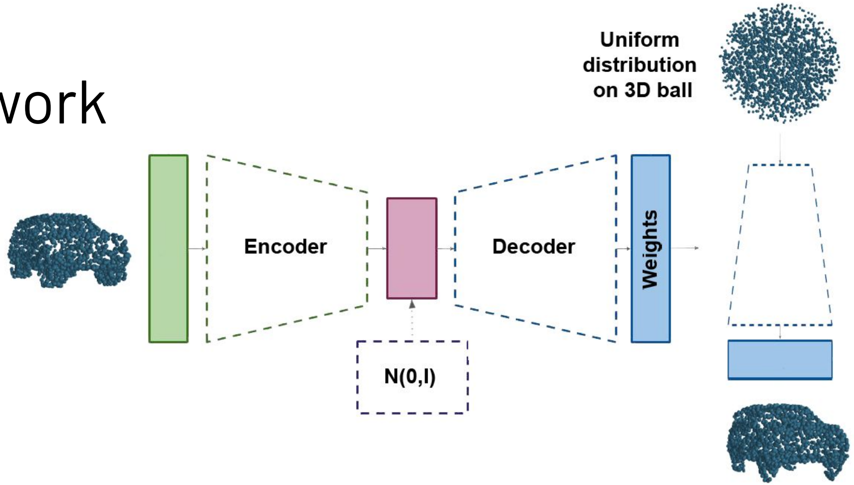
Related work: 3d Adversarial Autoencoders

- Adversarial Autoencoders adapted to point cloud data
  - Adversarial Autoencoders (Makhzani et al., ICLR 2016)
  - 3d Adversarial Autoencoders (Zamorski et al., CVIU 2019)
- Using PointNet as an Encoder
  - PointNet (Qi et al., CVPR 2017)
- Able to use an arbitrary prior
- Only generates a fixed number of points



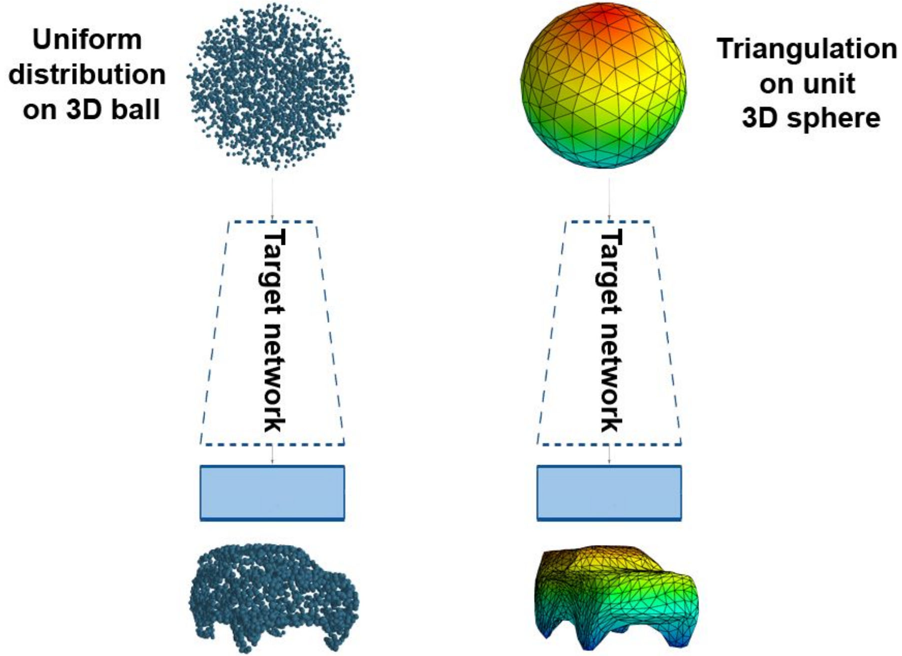
HyperCloud

- AAE architecture for hypernetwork
- PointNet as an Encoder
- Arbitrary prior on latent space
- Decoder produces weights for target network based on latent embedding
- Target network moves points from the uniform distribution on 3D ball to the 3D object



Easily extendible to meshes

- Use precomputed mesh instead of a point cloud
- Feeding vertices to the target network produces high-quality meshes
- No need for second mesh rendering



Experimental results

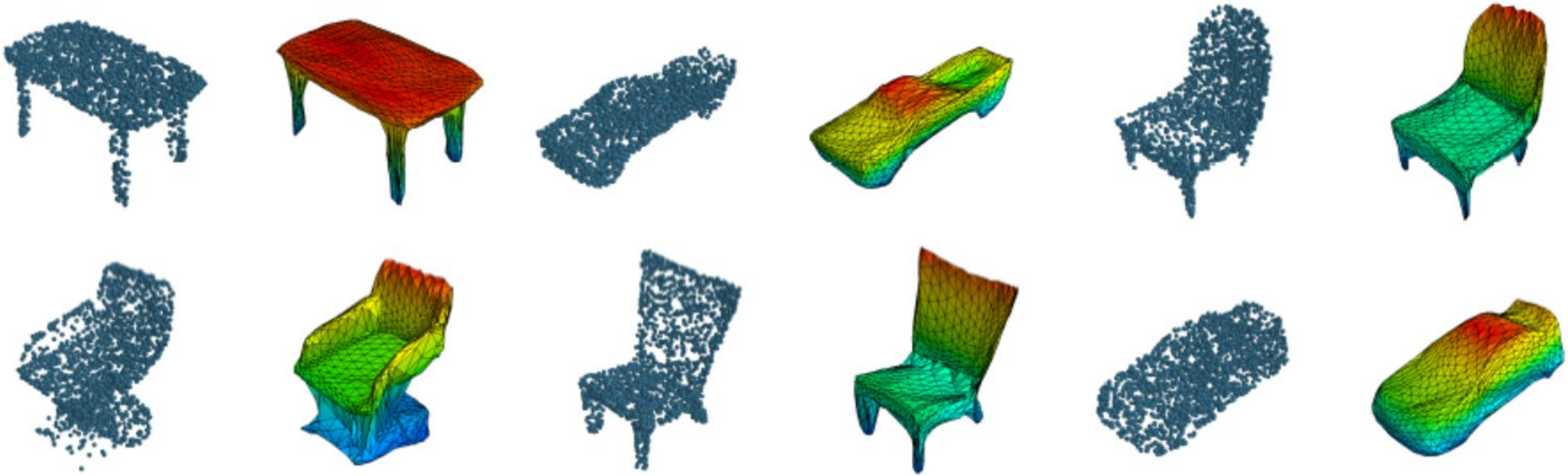


Figure: 3D point clouds and their mesh representations produced by HyperCloud

Evaluation

- Jensen-Shannon Divergence
  - The distance between two distributions
- Coverage
  - A portion of the reference data distribution that is covered by generated samples
- Minimum Matching Distance
  - Similarity of generated samples with respect to the reference set
- 1-Nearest Neighbour Accuracy
  - Are sample and reference test sets indistinguishable to simple classifier?

Experimental results

Table: Generation results (JSD ↓, MMD ↓, COV↑ , 1-NNA)

Class	Model	JSD	MMD		COV		1-NNA	
			CD	EMD	CD	EMD	CD	EMD
Airplane	r-GAN	7.44	0.261	5.47	42.72	18.02	93.58	99.51
	l-GAN (CD)	4.62	0.239	4.27	43.21	21.23	86.30	97.28
	l-GAN (EMD)	**3.61**	0.269	3.29	**47.90**	**50.62**	87.65	85.68
	PC-GAN	4.63	0.287	3.57	36.46	40.94	94.35	92.32
	PointFlow	4.92	**0.217**	**3.24**	46.91	48.40	**75.68**	**75.06**
	HyperCloud	4.84	0.266	3.28	39.75	43.70	93.80	88.95
	Training set	6.61	0.226	3.08	42.72	49.14	70.62	67.53
Chair	r-GAN	11.5	2.57	12.8	33.99	9.97	71.75	99.47
	l-GAN (CD)	4.59	2.46	8.91	41.39	25.68	64.43	85.27
	l-GAN (EMD)	2.27	2.61	7.85	40.79	41.69	64.73	65.56
	PC-GAN	3.90	2.75	8.20	36.50	38.98	76.03	78.37
	PointFlow	**1.74**	**2.42**	7.87	**46.83**	**46.98**	**60.88**	**59.89**
	HyperCloud	2.73	2.56	**7.84**	41.54	46.67	68.20	68.80
	Training set	1.50	1.92	7.38	57.25	55.44	59.67	58.46
Car	r-GAN	12.8	1.27	8.74	15.06	9.38	97.87	99.86
	l-GAN (CD)	4.43	1.55	6.25	38.64	18.47	63.07	88.07
	l-GAN (EMD)	2.21	1.48	5.43	39.20	39.77	69.74	68.32
	PC-GAN	5.85	1.12	5.83	23.56	30.29	92.19	90.87
	PointFlow	**0.87**	**0.91**	**5.22**	**44.03**	**46.59**	**60.65**	**62.36**
	HyperCloud	3.09	1.07	5.38	40.05	40.05	84.65	77.27
	Train set	0.86	1.03	5.33	48.30	51.42	57.39	53.27

Able to use EMD loss

Produce an arbitrary number of points

Experimental results

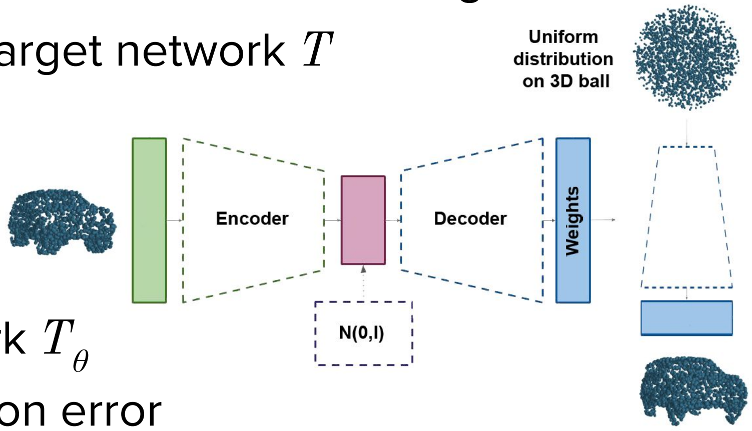
Table: Quality of representations by sampling from sphere (JSD ↓, MMD ↓, COV↑)

Class	Model	Sphere R	JSD	MMD		COV	
				CD	EMD	CD	EMD
Airplane	PointFlow	2.795	22.26	0.49	6.65	**44.69**	20.74
		3.136	26.46	0.60	6.89	39.50	19.01
		3.368	29.65	0.68	6.84	40.49	16.79
	HyperCloud	1.000	**9.51**	**0.45**	**5.29**	30.60	**28.88**
Chair	PointFlow	2.795	19.28	4.28	13.38	36.85	20.84
		3.136	22.52	4.89	14.47	32.47	17.22
		3.368	24.68	5.36	14.97	31.41	17.06
	HyperCloud	1.000	**4.32**	**2.81**	**9.32**	**40.33**	**40.63**
Car	PointFlow	2.795	16.59	1.60	8.00	20.17	17.04
		3.136	20.21	1.75	7.80	21.59	17.32
		3.368	24.10	1.96	8.35	18.75	17.04
	HyperCloud	1.000	**5.20**	**1.11**	**6.54**	**37.21**	**28.40**

Training details

- Use point cloud X as an input to the encoder E to obtain encoding z
- Based on z, we generate weights θ for the target network T
- Sample same number of points from the 3D prior as in X
- To obtain reconstruction X' pass sampled points through parameterized target network T_θ
- Calculate the loss consisting of reconstruction error and latent space regularization

$$L(X; E, D, P) = Err(X, D(E(X))) + Reg(E(X), P)$$



Experimental results

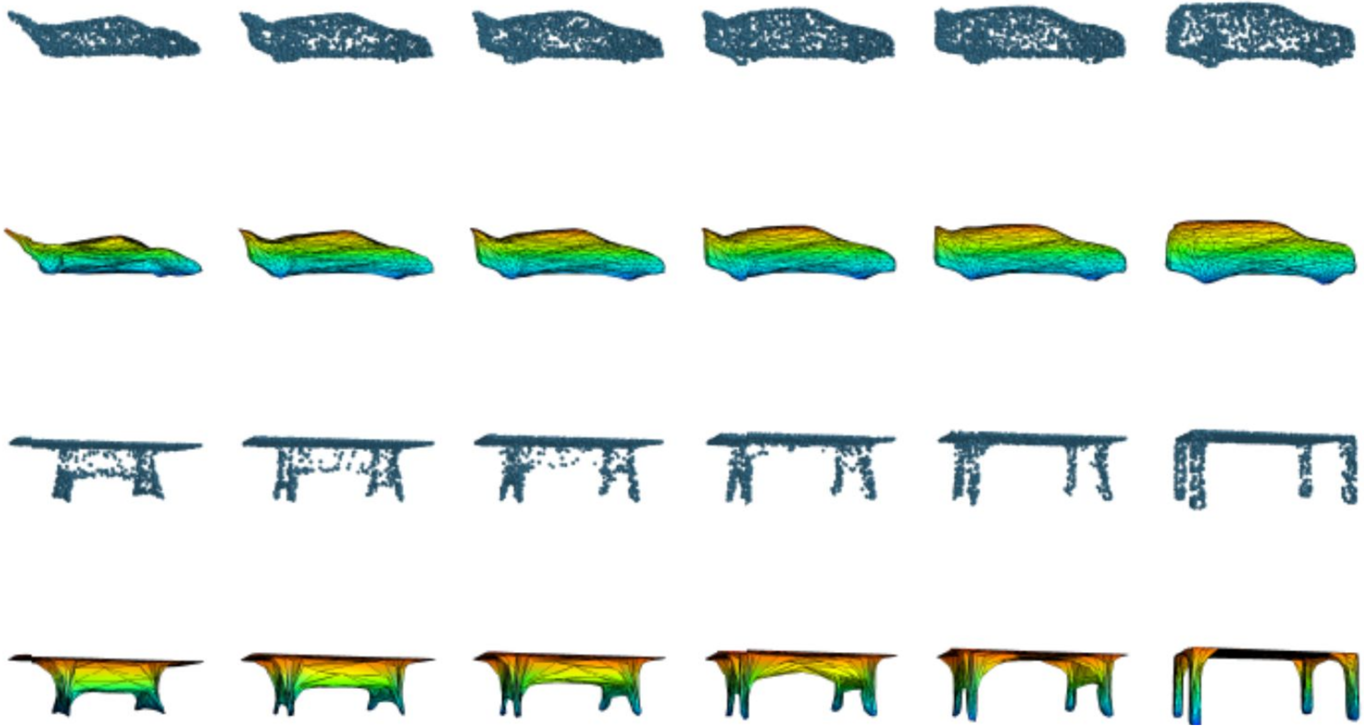


Figure: Interpolations between two 3D point clouds and their mesh representations

More experimental results

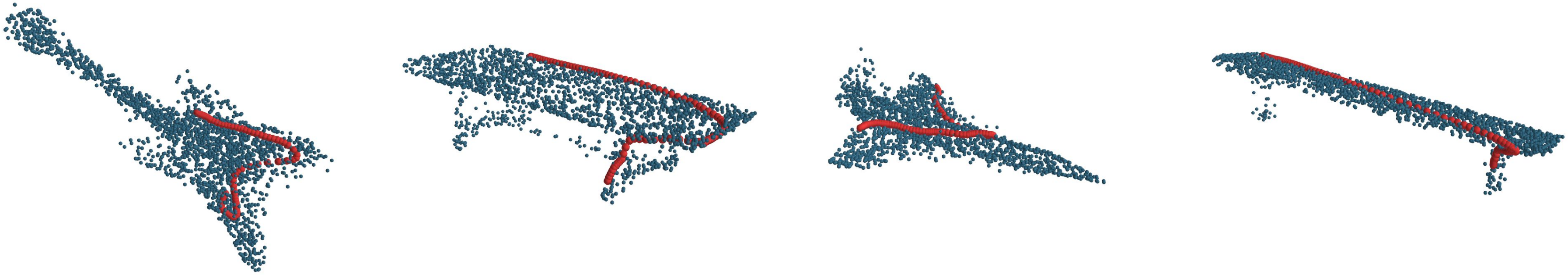


Figure: Interpolation between two points sampled from the 3D ball prior

Conclusion

- We present a novel method for generating 3D point clouds
- Our approach is able to work not only on point clouds, but also on 3D meshes
- We leverage the hypernetworks to obtain simple architecture and fast end-to-end training
- Our model is able to generate shapes consisting of an arbitrary number of points or vertices

Hypernetwork approach to generating point clouds

Przemysław Spurek[1], Sebastian Winczowski[1], Jacek Tabor[1], Maciej Zamorski[2 4], Maciej Zięba[2 4], Tomasz Trzciński[3 4]

[1] Jagiellonian University
[2] Wrocław University of Science and Technology
[3] Warsaw University of Technology
[4] Tooploox