Pyramid U-Network for Skeleton Extraction from Shape Points

Rowel Atienza
University of the Philippines
Electrical and Electronics Engineering Institute, Diliman, Quezon City, Philippines
rowel@eee.upd.edu.ph

Abstract

The knowledge about the skeleton of a given geometric shape has many practical applications such as shape animation, shape comparison, shape recognition, and estimating structural strength. Skeleton extraction becomes a more challenging problem when the topology is represented in point cloud domain. In this paper, we present the network architecture, PSPU-SkelNet, for TeamPH which ranked 3rd in Point SkelNetOn 2019 challenge [2]. PSPU-SkelNet is a pyramid of three U-Nets that predicts the skeleton from a given shape point cloud. PSPU-SkelNet achieves a Chamfer Distance (CD) of 2.9105 on the final test dataset. The code of PSPU-SkelNet is available at https://github.com/roatienza/skelnet.

1. Introduction

The knowledge about the skeleton provides a compact and intuitive representation of the shape for modeling, synthesis, compression, and analysis [2]. Practical applications of skeleton include shape animation, comparison, recognition, and estimating structural strength.

In 2D, shapes can be represented in terms of pixels or point clouds depending on the generating or acquisition device used. While image is the more common 2D shape format, point clouds are the natural outputs of laser scanning devices. Point clouds are simply an unordered set of points. For 2D shapes, a point cloud is a list of (x, y) coordinates with no fixed length. In this paper, we focus on extracting skeletons from 2D shape point clouds.

For the Point SkelNetOn 2019 competition [2], the ground truth skeletons are extracted from shape images using the method of [3, 8]. The candidate skeletons are produced by propagating a circle inside the shape. The potential skeletons are 2, 4 and 6-pixel Hausdorff distance away from the shape. The candidate that is most visually appealing is chosen. Alternative methods in skeleton extraction include determining the shape medial axis [17], continuous medial axis using an extended distance function [9], locus of midpoints of optimally corresponding boundary points using A* algorithm [10], and medial axis as singularities formed during a propagation from boundaries known as shock graphs and trees [14].

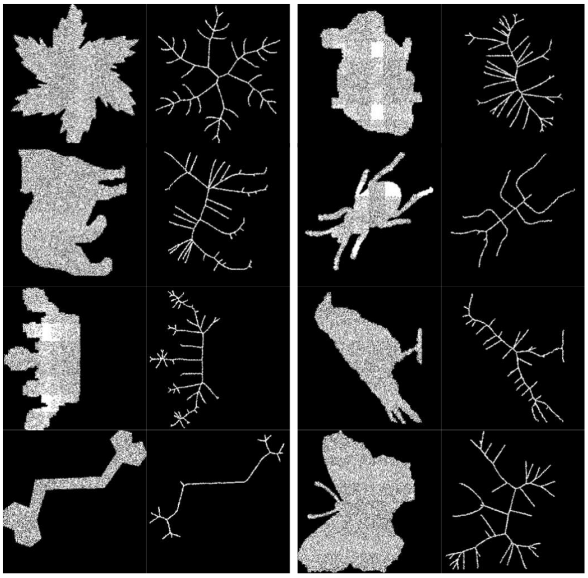

Figure 1. Example extracted shape and skeleton point clouds from train split.

Given the shape images, the shape point clouds ground truths are generated by first finding the boundary points. Once the boundary points have been identified, the shape point cloud is filled by points perturbed by noise from a uniform distribution. The skeletal point cloud is derived by choosing points from the shape point cloud that are near the image skeleton points and skeleton edges.

Given the shape and skeletal point cloud dataset [2, 1, 9, 15], we tried different network architectures, data and hyperparameter configurations, and loss functions. The best performing network architecture is obtained by assuming the point cloud as image forming data that can be processed using deep convolutional networks. More precisely we employed a pyramid of U-Nets. The pyramid network is in-

spired by PSPNet [16] for scene parsing. U-Net [13] is a known reliable network configuration for many segmentation and style transfer tasks . We call the network PSPU-SkelNet. The best performing results for our team is 2.9105 in terms of Chamfer Distance metric.

In the following sections, we describe the dataset, the metric of evaluation, our proposed network, and experimental results.

2. Dataset

The dataset of Point SkelNetOn 2019 competition [2, 1, 9, 15] for skeletal point cloud prediction is made of 1219 shapes with corresponding ground truth skeletons for the train split. This is initially evaluated on 242 shape point clouds during development phase. The final test split is made of 266 shape point clouds. Figure 1 shows example shapes and skeletons from the train split.

3. Metric

The leaderboard score is measured in terms of Chamfer Distance (CD). Given two point clouds $p_1 \in S_1 \subseteq \mathbb{R}^2$ and $p_2 \in S_2 \subseteq \mathbb{R}^2$:

$$CD = \frac{1}{|S_1|} \sum_{p_1 \in S_1} \min_{p_2 \in S_2} \|p_1 - p_2\|_2 + \frac{1}{|S_2|} \sum_{p_2 \in S_2} \min_{p_1 \in S_1} \|p_1 - p_2\|_2 \quad (1)$$

For the first summation, given a point p_1 in the first point cloud S_1, find the nearest corresponding point p_2 in the second point cloud S_2. The same process is done from p_2 to p_1 for the second summation. S_1 and S_2 do not necessarily have the same number of points. CD is differentiable and is computationally efficient.

4. Pyramid U-Net

In our experiments, the best performing network architecture for extracting skeletons from shape point clouds is a pyramid of U-Nets as shown in Figure 2. The shape point clouds are deserialized into shape images for fast and efficient 2D convolution processing. The output skeleton images are serialized back into skeleton point clouds for performance benchmarking. Sigmoid predictions greater than $threshold = 0.5$ are considered skeletal point clouds.

In PSPU-SkelNet, we used three U-Nets with different depths based on strides. PSPU-SkelNet has $strides = 2, 4,$ and 8. Each U-Net extracts features at different receptive field sizes. The U-Net structure is inspired by $pix2pix$ [6] for image to image translation. Instead of batch normalization, we utilized instance normalization since it has a better performance in style transfer problems [4].

The outputs of each left branch of U-Net are feature maps that are upsampled and stacked together on the right branch. This configuration is inspired by PSPNet [16] used in scene parsing. PSPNet network processes feature maps at different coverage and depths.

PSPU-SkelNet uses binary cross-entropy loss. The optimizer is Adam [7] with initial learning rate of $1e-3$. The batch size is set to 8.

5. Experimental Results

Since the dataset is small, we performed data augmentation before training. Each pair of shape and skeleton point cloud is subjected to random translation, rotation, scaling, and flipping. The resulting total amount of data is $26x$ the original train dataset size.

We trained PSPU-SkelNet for 200 epochs starting with a learning rate of $1e-3$ and decreased to $1e-4$ at 60 epochs and $1e-5$ at 120 epochs. Each epoch takes about 20mins to complete on a single NVIDIA GTX 1080Ti GPU.

As of submission deadline our team, TeamPH, ranks 3rd with a prediction score of 2.9105 on the test data split as shown in Table 1. Example prediction results are shown in Figure 3. Visually, majority of the predictions are near the expected skeletons. The network struggles on point clouds with small number of samples from the train datasset such as those under a specific device category.

5.1. Things that we tried but did not work out

There are many network, data, and hyperparameter configurations that we tried. We enumerate the failed experiments as follows:

- We tried the native point cloud data format and built the pyramid U-Net using 1D convolution similar to PointNet [11]. We used a differentiable CD loss function. Since the point cloud has no fixed length, we padded with $(0, 0)$ to make the length constant at 1024×12. After training the network, points near $(0, 0)$ were removed. The resulting skeleton predictions contain many outlier point clouds. Since $(0, 0)$ is also a valid point that may introduce spurious data, we tried resampling the point cloud to complete the 1024×12 points. This technique did not help.

- We tried using MAE instead of binary cross-entropy. The resulting skeleton point clouds are thicker resulting to lower scores.

- Similar to $pix2pix$ [6], we tried using dropout after every encoder and decoder layer in U-Net. This resulted to lower prediction scores.

- Higher number of filters in U-Net results to a huge number of parameters that can easily approach 100M.

Figure 2. Network architecture of PSPU-SkelNet. f is number of filters. s is strides. k is kernel size (default is 3). $Conv2D$ is 2D convolution while $Conv2DT$ is transposed 2D convolution. IN is instance normalization. $ReLU$ is rectified linear unit.

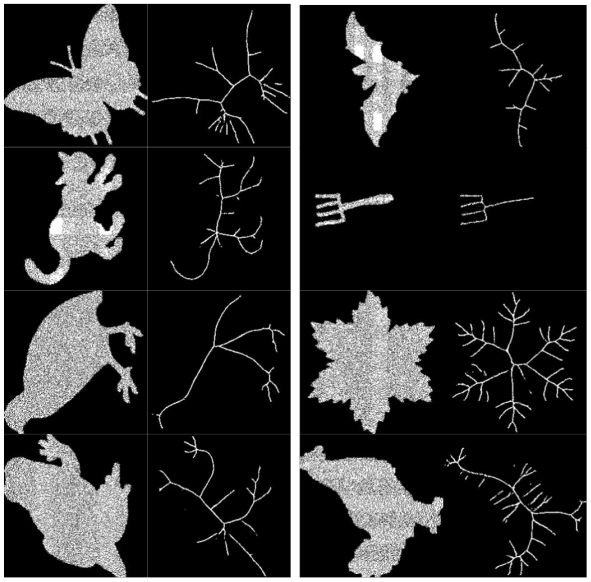

Figure 3. Example shape and skeleton point clouds from test split.

This network became impractical to train in a reasonable amount of time. Our current network is already big at 57M parameters. We did not perform pruning or apply compression due to resource and time constraints.

Table 1. Point SkelNetOn Leaderboard as of April 3, 2019

Rank	Username	Team Name	Prediction Score
1	digitalspecialists	RG	1.8706
2	Vvf	Prisdl	2.6789
3	rowel	TeamPH	2.9105

- We tried using GAN [5, 12] to improve the CD score. The PSPU-SkelNet is treated as a generator. A separate discriminator network was used with the standard binary cross-entropy loss. The performance did not improve. Instead, many outlier point clouds were introduced.

6. Conclusion

We described a proposed network as a solution to the problem for skeletal point cloud prediction using a pyramid of U-Net. Quantitative results rank TeamPH as 3rd in terms of prediction scores.

Acknowledgement

Supported in part by DOST-ERDT FRDG and CHED-PCARI Project IIID-2016-005.

References

[1] Alexander M Bronstein, Michael M Bronstein, Alfred M Bruckstein, and Ron Kimmel. Analysis of two-dimensional non-rigid shapes. *International Journal of Computer Vision*, 78(1):67–88, 2008.

[2] Ilke Demir, Camilla Hahn, Kathryn Leonard, Geraldine Morin, Dana Rahbani, Athina Panotopoulou, Amelie Fondevilla, Elena Balashova, Bastien Durix, and Adam Kortylewski. SkelNetOn 2019 Dataset and Challenge on Deep Learning for Geometric Shape Understanding. *arXiv e-prints*, 2019.

[3] Bastien Durix, Sylvie Chambon, Kathryn Leonard, Jean-Luc Mari, and Géraldine Morin. The propagated skeleton: A robust detail-preserving approach. In *International Conference on Discrete Geometry for Computer Imagery*, pages 343–354. Springer, 2019.

[4] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015.

[5] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[6] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[7] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[8] Aurélie Leborgne, Julien Mille, and Laure Tougne. Extracting noise-resistant skeleton on digital shapes for graph matching. In *International Symposium on Visual Computing*, pages 293–302. Springer, 2014.

[9] Kathryn Leonard, Geraldine Morin, Stefanie Hahmann, and Axel Carlier. A 2d shape structure for decomposition and part similarity. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 3216–3221. IEEE, 2016.

[10] Tyng-Luh Liu and Davi Geiger. Approximate tree matching and shape similarity. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 1, pages 456–462. IEEE, 1999.

[11] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 1(2):4, 2017.

[12] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *ICLR*, 2016.

[13] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[14] Thomas B. Sebastian, Philip N. Klein, and Benjamin B. Kimia. Recognition of shapes by editing their shock graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:550–571, 2004.

[15] Thomas B Sebastian, Philip N Klein, and Benjamin B Kimia. Recognition of shapes by editing their shock graphs. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (5):550–571, 2004.

[16] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.

[17] Song Chun Zhu and Alan L Yuille. Forms: a flexible object recognition and modelling system. *International journal of computer vision*, 20(3):187–212, 1996.