HyperSDFusion: Bridging Hierarchical Structures in Language and Geometry for Enhanced 3D Text2Shape Generation

Supplementary Material

Our main paper introduced HyperSDFusion for text-to-shape generation, which explores how to bridge hierarchical structures in language and geometry. In this supplemental document, we provide more detailed information about our method and experiments.

6.1. The Details of MERU

In our hyperbolic text-image encoder introduced in Section 4.2, we employ the text encoder of MERU [10] to learn text sequential features embedded with hierarchical multi-modal features. In this supplemental document, the details of MERU are described.

MERU is a large-scale contrastive image-text model that yields hyperbolic representations capturing the visual-semantic hierarchy. As shown in Figure 3, MERU consists of two separate text-image encoders, feature projection, and contrastive loss. The text encoder is multiple layers of transformer encoder blocks. The image encoder is the small Vision Transformer [7]. The feature projection is implemented by the Exp function, which projects features to hyperbolic space. Under the supervision of the designed contrastive loss (a contrastive loss and an entailment loss), MERU enforces partial order relationships between paired text and images. For more details, please refer to [10].

6.2. The Details of Text Graph Building

The first step of our hyperbolic text-graph convolution module is text-graph initialization. In this supplemental document, we will explain more details of text-graph building.

As mentioned in Section 4.2, we process texts using spaCy [17], and obtain a syntax tree. The syntax tree is represented as a text graph by traversing the child nodes of the tree. The algorithm for the traversal process is elaborated in Alg 1.

Algorithm 1: Framework of The Transformation of Tree-to-graph.

Input: The syntax tree with n nodes: $T_G = \{t_{i,G} | i = 0, .., n-1\}$.
Output: The adjacent matrix of a text graph: $M_{n \times n}$

$i = 0$;
while $t_{i,G}$ *in* T_G **do**
  $M_{i,i} = 1$;
  foreach *child in* $t_{i,G}$.*child* **do**
    $j = child.index$;
    if $j <$ *n-1* **then**
      $M_{i,j} = 1$;
      $M_{j,i} = 1$;

6.3. The Details of Hyperbolic Hierarchical Loss

As mentioned in Section 4.3, we proposed a hyperbolic hierarchical loss to supervise the hierarchical structure of 3D feature space between deep and shallow features, f_h, f_m, f_s, which are the output of the 3D U-Net at three scales. We process these features by our hyperbolic hierarchical loss, followed by steps shown in Alg 2.

Algorithm 2: Framework of Hyperbolic Hierarchical Loss.

Input: Deep features: f_h;
Middle features: f_m;
Shallow features: fs;
The dimensions of hyperbolic space: C.
Output: Computed loss: L.

f_h = MLP(Pooling(f_h));
f_m = MLP(Pooling(f_m));
f_s = MLP(Pooling(f_s));
foreach f *in* $\{f_h, f_m, f_s\}$ **do**
  f = Exp(Möbius(f));
ball = PoincareBall(c=1.0, dim=C);
d_1 = ball.dist0(f_h);
d_2 = ball.dist0(f_m);
d_3 = ball.dist0(f_s);
L = max(0, -d_2+d_1) + max(0, -d_3 + d_2).

6.4. More Qualitative Results on Capturing Text-shape Hierarchy

In Section 5.3, we have given some results to present our advantage of capturing the text-shape hierarchy. In this supplemental document, we provide more qualitative results on capturing text-shape hierarchy.

The hierarchy of text feature We visualize 2D text embeddings of 1000 random training samples in Figure 7. The dot color represents the length of the text. The light blue dot refers to the short text, that is general text without detailed information, like "a chair". The dark blue refers to the long text, that is detailed text, like "The silver and brown color iron chair with four legs and sponge.". As illustrated in Figure 7 (a), the 2D text embeddings learned by SDFusion [8] are cluttered because the text encoder of SDFusion [8], BERT in Euclidean space [11], cannot capture the

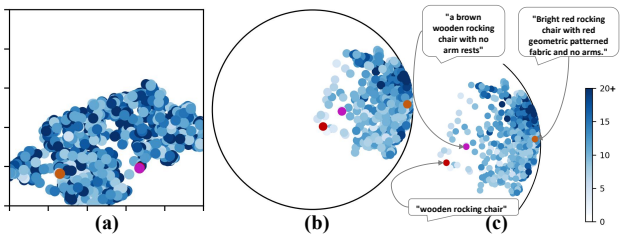

Figure 7. UMAP visualization of 2D text embeddings of 1000 random training samples. The color bar indicates the length of the text. (a): 2D text embeddings learned by SDFsuion [8] in Euclidean space. (b): 2D text embeddings learned by our method in hyperbolic space. (c) is the magnified view of (b).

text-shape hierarchy. In contrast, it can be observed from Figure 7 (c) that the text length of text embeddings learned by our hyperbolic text-image encoder increases along the radius. It represents that features of general texts are close to the center point, and features of detailed text are near the boundary, exhibiting a hierarchical structure in hyperbolic space. Furthermore, we highlight a sample of text hierarchy in Figure 7, the red point refers to a general text, "wooden rocking chair", a pink point refers to a middle-level text, "a brown wooden rocking chair with no arm rests", and an orange point refers to a more detailed text, "Bright red rocking chair with red geometric patterned fabric and no arms.". It can be observed that the text embeddings of these points in Figure 7 (a) do not follow the hierarchical structure, while the text embeddings of these points in Figure 7 in Figure 7 exhibits the hierarchical structure.

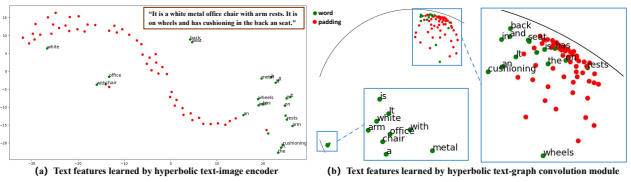

Figure 8. Text features learned by our HTIE and HTGC modules.

Learned two kinds of text features Employing these two kinds of text features aims to leverage both the inherent sequential property and linguistic structures of text. Depicted in Figure 8, the feature distribution in Figure 8(**a**) showcases its sequential nature, while features in Figure 8(**b**) are more consistent with linguistic structure correlation.

3D shape feature visualization. We also illustrate the feature distribution in Euclidean and hyperbolic space, as shown in Figure 9. It is observed that features in Euclidean space do not exhibit a tree-like hierarchical structure, conversely to those in hyperbolic space, which expand from the gray origin to the deep features in blue, the middle features in green, and finally to the shallow features in orange. It

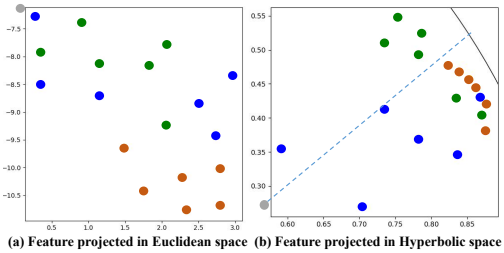

Figure 9. Features of 3D shape projected in the Euclidean space and hyperbolic space by Umap. The blue dot signifies deep features, green denotes middle features, and orange represents shallow features. The blue line is the radius of the Poincaré Ball.

Text	HMD↓ SDFusion [8]	Ours
"wood chair"	-	-
"wood square chair"	1.84	**0.53**
"wooden color square type wooden chair 4 leg"	0.40	**0.04**
"four leg chair made of wood square base and good comfort for back"	0.87	**0.03**
"modern chair"	-	-
"Modern silver and gray office chair."	2.57	**0.24**
"Modern office chair with three legs made of metal and fibre made black seat."	1.18	**0.34**
"It is a soft sofa"	-	-
"a soft sofa chair with 4 stand support of grey color"	2.35	**0.29**
"A grey cushioned sofa with a curved back rest and four thin legs."	**0.82**	0.93
"furniture"	-	-
"square, folding, furniture to sit on, black and beige"	1.82	**0.32**
"brown, square, sitting furniture with a hole designe on the arms and back"	0.67	**0.29**
"couch"	-	-
"This couch is blue in color and has four legs."	2.76	**0.48**
"A gray couch heavily-cushioned with very tall backrest and stubby legs."	**1.38**	1.63
"Bamboo chair"	-	-
"a wooden chair colored like bamboo, with a steel frame."	2.24	**0.45**
"A BAMBOO BORDER ROUND BASED SEATING ARM LESS CHAIR WITH CUSHIONS DECORATED IN POLKA DOT MATERIAL"	1.42	**0.23**

Table 6. The result of comparing the performance of capturing the text-shape hierarchical structure.

indicates the denoiser supervised by our hyperbolic hierarchical loss guarantees the tree-like hierarchical structure of 3D shape.

More results and analysis for text-shape hierarchy. In Table 4, we have provided a sample of hierarchical text, and the HMD between the generated shapes. In Table 6, we enumerate more samples to demonstrate our performance of capturing text-shape hierarchy. Moreover, We also provide more visualizations for capturing text-shape hierarchy in Figure 10, Figure 11, and Figure 12. It can be observed that 3D shapes generated by our method exhibit a hierarchy from general texts to detailed texts. In contrast, the shapes generated by SDFusion [8] from general texts to detailed texts do not correlate.

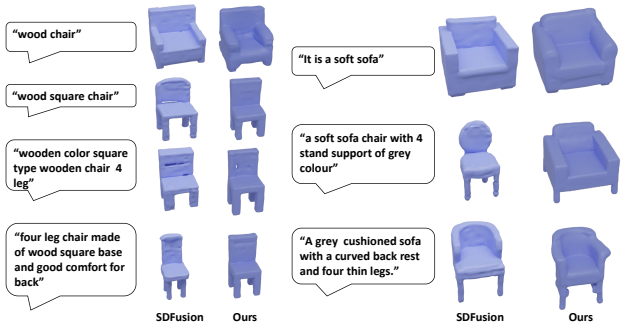

Figure 10. More visualizations for capturing text-shape hierarchy.

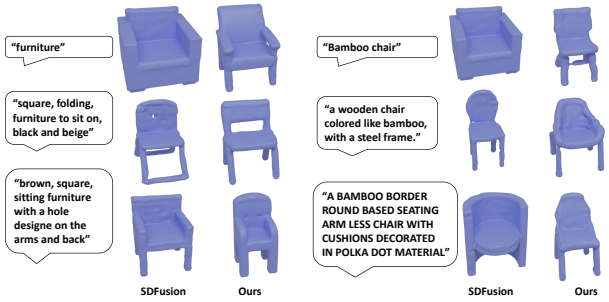

Figure 11. More visualizations for capturing text-shape hierarchy.

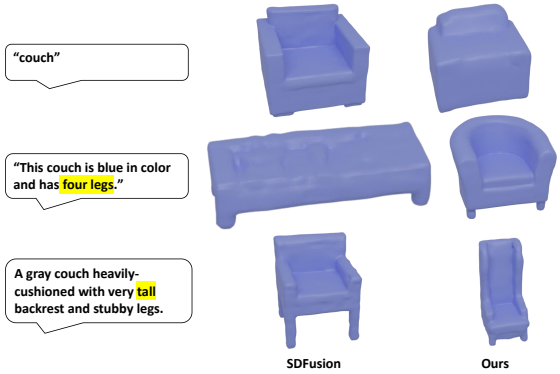

Figure 12. More visualizations for capturing text-shape hierarchy.