Variational Point Encoding Deformation for Dental Modeling

Johan Ziruo Ye*,†
Johan.Ye@3shape.com

Thomas Ørkild†
Thomas.Orkild@3shape.com

Peter Lempel Søndergaard†
Peter.Soendergaard@3shape.com

Søren Hauberg*
sohau@dtu.dk

*Technical University of Denmark †3Shape

Abstract

Digital dentistry has made significant advancements in recent years, yet numerous challenges remain to be addressed. In this study, we release a new extensive dataset of tooth meshes to encourage further research. Additionally, we propose Variational FoldingNet (VF-Net), which extends FoldingNet to enable probabilistic learning of point cloud representations. A key challenge in existing latent variable models for point clouds is the lack of a 1-to-1 mapping between input points and output points. Instead, they must rely on optimizing Chamfer distances, a metric that does not have a normalized distributional counterpart, preventing its usage in probabilistic models. We demonstrate that explicit minimization of Chamfer distances can be replaced by a suitable encoder, which allows us to increase computational efficiency while simplifying the probabilistic extension. Our experimental findings present empirical evidence demonstrating the superior performance of VF-Net over existing models in terms of dental scan reconstruction and extrapolation. Additionally, our investigation highlights the robustness of VF-Net’s latent representations. These results underscore the promising prospects of VF-Net as an effective and reliable method for point cloud reconstruction and analysis.

1 Introduction

Recent advances and adoption of intraoral scanners in dentistry have led to a large number of meshes of patients’ teeth available at micrometer resolution. These 3-dimensional models are used for surgical planning, tooth crown generation, tooth wear estimation, etc. By treating such meshes as point clouds, we gain a computationally efficient representation of the shape and topology of patients’ teeth through a sparse set of points. However, consequently any statistical modeling of point clouds must be invariant to any reordering and variability in cardinality that may occur.

Our research is motivated by the need to analyze, search, and organize large collections of such dental scans. The sensitivity of the dentists tasks necessitates robustness to noisy data and feedback on model uncertainty to the responsible dentist. The foundation of our paper is a new dataset, the FDI 16 Tooth Dataset, which provides a large collection of dental scans. Our primary objective is to learn useful and reliable representations of this data. However, in our paper, we also highlight other crucial tasks such as shape completion of the sides of the tooth unable to be scanned by the intraoral scanner, as well as shape completion of areas previous obstructed by braces or other orthodontic devices.

1Code at https://github.com/JohanYe/VF-Net

2Available here.

Preprint. Under review.
Figure 1: The top row displays four different teeth from the FDI 16 tooth dataset, notably these meshes are irregular and vary in resolution. The bottom row displays the reconstructed meshes from FoldingNet. Its folding artifacts prevent model usage in computational dentistry. Furthermore, the left two teeth in the bottom row have their facets highlighted, while the right two teeth do not.

**Point cloud representation learning for teeth.** As most teeth both move and degrade continuously with time, it is reasonable to seek a continuous vectorial representation when organizing extensive collections of dental scans. This motivates the development of autoencoder-style representation learning models for point cloud data [1, 2]. Such models form an encoder-decoder pair that reconstructs a given input point cloud from a learned vectorial representation by minimizing a reconstruction error. Since we generally do not have correspondences between points in two given clouds, pairwise Euclidean distances are inapplicable and instead permutation invariant metrics are used. The most common choice is the **Chamfer distance** (CD) [3] defined as

$$
CD(X, Y) = \frac{1}{|X|} \sum_{x \in X} \min_{y \in Y} \|x - y\|_2 + \frac{1}{|Y|} \sum_{y \in Y} \min_{x \in X} \|y - x\|_2,
$$

for point clouds $X$ and $Y$. Although this choice of metric solves the invariance problem, it poses a new one: *The Chamfer distance (1) does not readily lead to a likelihood, preventing its use in probabilistic models.* For instance, when used in the Gaussian distribution, the function $X \mapsto 1/c \exp(-CD^2(X, \mu))$ cannot be normalized to have unit integral due to the explicit minimization in Eq. 1. For our application, we need robustness to noise and general quantification of uncertainty, thus we consider the lack of an explicit likelihood for point clouds to be detrimental. Unsupervised probabilistic representation learning has been shown to be beneficial across many tasks including generative modeling [4–6], out-of-distribution detection [7, 8], handling missing data [9] and more.

**In this paper,** we propose a new architecture that allows us to sidestep the use of Chamfer distances, which, in turn, allows for straightforward constructions of models akin to **variational autoencoders (VAEs)** [4, 5]. We call the resulting model the **Variational FoldingNet** (Sec. 3), as it bridges FoldingNet [2] and VAEs. A key aspect of our model is that it avoids the use of Chamfer distances, and instead relies on an encoder. Moreover, we contribute a new dataset of dental scans (Sec. 2) that contain the first maxillary molar tooth on the right side of the upper jaw — one of the most common teeth to receive dental treatment/restoration. Using this dataset, we explore keys tasks, such as shape completion in cases where neighboring teeth obstruct the view or shape completion impeded by orthodontic treatment. We also showcase the potential for future tasks in representation learning and style transfer. Finally, we demonstrate that for dental scans, our model performs superior or competitively on various standardized generative modeling tasks when compared to current state-of-the-art models (Sec. 5).

### 2 The FDI 16 Tooth Dataset

To improve state-of-the-art modeling of dental scans, we have gathered a new extensive dataset, which will be released alongside this paper. The FDI 16 dataset is a collection of 7,732 irregular meshes of the right-side first maxillary molar tooth\(^3\) that were collected fully anonymized from intraoral scans by 3Shape\(^4\). Each tooth in the FDI 16 Tooth dataset algorithmically segmented from a

---

\(^3\)Technically referred to as FDI 16 according to the ISO 3950 notation.

\(^4\)https://www.3shape.com/
scan of an upper jaw. These meshes are from patients undergoing aligner treatment, which likely introduces a bias towards younger individuals, who generally have fewer restorations and dental problems. Additionally, aligner attachments may be found on teeth. Unlike most existing 3D datasets, the scanned teeth constitute open meshes and have clear boundaries. The top row of Fig. 1 shows examples of such meshes.

The FDI 16 teeth meshes were collected using TRIOS scanners, primarily the TRIOS 3 model. Each scan is represented as an irregular polygonal mesh. These meshes all share highly similar topology, so the main differences between them are in their shapes. All teeth have clearly defined boundaries and are consequently open with no representation of the interior object volume. Our analysis is based on point clouds sampled from the meshes for computational efficiency. We have made both the meshes and point clouds publicly available. All teeth have been rotated to ensure that the $x$-axis is turned towards the neighboring tooth (FDI 17), while the $y$-axis points in the occlusal direction (direction of the biting surface). Finally, the $z$-axis is given by the cross-product to ensure a right-hand coordinate system. The scale of the data is in millimeters.

An initial study. Dental scans can be used for a variety of tasks in different research fields. In this study, we initiate our investigation by focusing on the reconstruction of teeth through latent variable modeling. An obvious candidate model for this task is FoldingNet [2], which reconstructs the original point cloud by deforming points from a 2D plane. This approach aligns well with the topology of the FDI 16 dataset, as no tearing of the latent grid should be required [10]. The bottom row of Fig. 1 shows reconstructions from FoldingNet. Here it is evident that FoldingNet introduces large gaps in the reconstructed meshes, which we consistently observe across meshes. These artifacts, in practice, prevent its usage in computational dentistry. Moreover, we demonstrate VF-Net’s ability to extrapolate previously unseen regions of the mesh without any additional training. This is particularly valuable since obtaining paired data of teeth with an obstructed view and unobstructed view is exceedingly rare, making training specialized methods difficult. We performed a more elaborate study of FoldingNet and other baseline methods in Sec. 5.

3 Variational Inference on Point Clouds

Notation. We denote vectors in lower case bold, e.g. $\mathbf{x}$, and matrices in upper case bold, e.g. $\mathbf{X}$. Point clouds are denoted as matrices, e.g. $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$ with $\mathbf{x}_n \in \mathbb{R}^3$. We let $\mathcal{G} = \{g_i\}_{i=1}^I$ denote a regular two-dimensional grid, where $I$ is the number of cells in the grid. Furthermore, $\text{proj}_M(\mathbf{x})$ represents the projection of a point $\mathbf{x} \in \mathbb{R}^3$ onto the two-dimensional surface $M$. Finally, $\mathcal{N}(\mathbf{x} \mid \mu, \sigma^2)$ denotes the density of the normal distribution with mean $\mu$ and variance $\sigma^2$.

Background: Variational autoencoders. The variational autoencoder (VAE) and the traditional autoencoder both attempt to identify a mapping between input space $\mathbf{x} \in \mathcal{X}$ and feature space $\mathbf{z} \in \mathcal{Z}$. The traditional autoencoder does so by minimizing the reconstruction error, $\mathcal{E} = ||\mathbf{x} - \hat{\mathbf{x}}||^2$. The variational autoencoder instead attempts to parameterize the density $p(\mathbf{x})$ through the latent variable $\mathbf{z}$ [4, 5]. The latent variable is given a non-informative prior, $p(\mathbf{z}) = \mathcal{N}(\mathbf{z} \mid 0, 1)$, which is marginalized
alongside an explicit conditional data likelihood \( p(x|z) \) to form the evidence \( p(x) = \int p(x|z)p(z)dz \). This integral is intractable and therefore does not constitute a practical loss function for training. Instead, an evidence lower bound (ELBO) \( \mathcal{L} \) is commonly used for training,

\[
p(x) \geq \mathcal{L}(x) = \mathbb{E}_{q(z|x)}[\log p(x|z)] - \text{KL}(q(z|x) || p(z)),
\]

where KL denotes the Kullback-Leibler divergence (KLD) and \( q(z|x) \) is any distribution. The bound is both tight and maximized when \( q(z|x) = p(z|x) \), where the right-hand side denotes the true latent posterior. In practice, both \( p(x|z) \) and \( q(z|x) \) are parameterized by neural networks. These serve similar purposes to the decoder and encoder pair of a classic autoencoder [1].

**Background: PointNet and FoldingNet.** As stated earlier, point clouds are sets of points with varying size and arbitrary order, and models thereof should unaffected by such changes to the point cloud. Therefore, one of the primary approaches to address such data is to develop neural networks that exhibit invariance changes in cardinality and permutation [2, 11]. Unfortunately, when it comes to the variational autoencoder this is not possible with current designs. A variational autoencoder that exhibit invariance changes in cardinality and permutation [2, 11]. Unfortunately, when it comes to the variational autoencoder this is not possible with current designs. A variational autoencoder outputs a distribution for each element in which the corresponding input element is evaluated [4, 5]. This translates to a distribution per input point. However, most current permutation-invariant neural networks do not track which input point each output point corresponds to.

FoldingNet achieves this invariance by using a PointNet-like encoder, \( e \), which entails utilizing multi-layer perceptrons (MLPs) that operate independently on each point of the point cloud. The folding-based decoder, \( \mu : \mathbb{R} \times \mathbb{R}^2 \rightarrow \mathbb{R}^3 \), is composed of two MLPs. These are applied to the latent code, \( z \), concatenated with each point in the chosen latent shape, \( G \), which in our case is the two-dimensional planar patch \([-1, 1]^2 \) [2]. The folding of the planar patch, \( \{\mu(g_i)\}_{i=1}^N \), is determined by the parameter vector \( z \) predicted by the PointNet encoder \( e \). Both the encoder \( e \) and the decoder \( \mu \) are jointly trained to minimize the reconstruction error

\[
\mathcal{E} = \sum_{n=1}^N \| x_n - \text{proj}_G(x_n) \|^2,
\]

where \( \text{proj}_G : \mathbb{R}^3 \rightarrow \mathbb{R}^3 \) denotes the projection of a point \( x \) onto the surface spanned by \( S = \mu(G) \).

\[
\text{proj}_G(x) = \mu(\hat{g}) \quad \text{where} \quad \hat{g} = \arg\min_{g \in \hat{G}} \| x - \mu(g) \|^2.
\]

This creates a permutation invariant and cardinality invariant autoencoder. FoldingNet approximates this projection during training using Chamfer distances (Eq. 1).

### 3.1 The Variational FoldingNet

Initially, we will outline the generative process of our proposed Variational FoldingNet (VF-Net) and then discuss approximate inference and training. Let \( p(z) \) be a Normalizing flow prior over the parameters describing the shape of an object [12]. Similarly to FoldingNet, we constrain our flat mesh grind \( G \) to be within the planar patch \([-1, 1]^2 \). This grid is, as in FoldingNet, subsequently deformed according to \( z \). Let \( g \in G \) denote a point on this grid, then the corresponding three-dimensional point \( x \) is defined to be distributed as \( p(x | z, g) = \mathcal{N}(x|\mu(z, g), \sigma^2(z, g)I) \), where \( \mu : \mathbb{R} \times \mathbb{R}^2 \rightarrow \mathbb{R}^3 \) and \( \sigma^2 : \mathbb{R} \times \mathbb{R}^2 \rightarrow \mathbb{R}^3 \) are neural networks. In this model, new samples can be generated by first sampling \( z \) and then mapping the grid points through \( \mu \) and \( \sigma \),

\[
x = \mu(z, g) + \sigma(z, g) \cdot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I).
\]

This process defines the likelihood of the observed data as \( p(x) = \int p(x | z)p(z)dz \) which gives a training objective. Unfortunately, the integral is generally intractable, and approximations are necessary. Following conventional variational inference [4, 5], a lower bound on \( p(x) \) is given by

\[
\mathcal{L}(x) = \mathbb{E}_{q(z|x)}[\log p(x | z)] - \text{KL}(q(z | x) \| p(z)),
\]

where \( q(z | x) \) is any approximation to the posterior \( p(z | x) \). To evaluate the above equation, we establish a 1-to-1 correspondence throughout the entire network, by introducing a projection \( \text{proj}_G(x) : \mathbb{R}^3 \rightarrow G \), where \( \text{proj}_G \) is a neural network (see Fig. 2). Again, we optimize Eq. 4, where the introduction of \( \text{proj}_G \) means that \( G \) is no longer independent and can now be optimized together with \( \mu \). This
latent point encoding gives us a 1-to-1 mapping throughout the network, allowing for evaluation using
the ELBo (Eq. 6). In which we utilize multivariate normal distribution with isotropic variance as the re-
construction term. This is a novel method of evaluation for 3D reconstruction networks, which avoids
the computationally expensive Chamfer distance. In supplementary Fig. S1, we demonstrate that Eu-
clidean distances serve as a suitable substitute for Chamfer distances. We observe close alignment be-
tween the two metrics with the Euclidean distances forming an upper bound that progressively tightens
as reconstruction precision increases. Other permutation invariant networks lack a 1-to-1 connection
between input and output, making such evaluation impossible. Instead, they rely on Chamfer distances
(1) [2, 13, 14], but no suitable normalization constant can be derived for probabilistic distribution us-
ing Chamfer distances. Consequently, probabilistic evaluation using chamfer distances is impossible.

4 Related work

We focus on point cloud representations of 3D objects [2, 15, 16], but there are many alternatives,
including voxel grids [17, 18], multi-angle inference [19, 20], and meshes [13, 21, 22]. A major
paradigm in point cloud processing neural networks is to remain permutation and cardinality invariant.
When it comes to reconstruction, this often results in designs without a 1-to-1 mapping between
inputs and outputs [2, 13]. This becomes an obstacle in adopting the variational autoencoder to point
clouds. Accordingly, other methods have become prominent, including GANs, diffusion models, and
traditional autoencoders [10, 13, 15, 16, 23–26].

Another attempt to design a variational autoencoder for point clouds is SetVAE [14], which uses
transformers to process sets including point clouds. One of their novelties is the introduction of a
latent space with an enforced prior inside the transformer block. These transformer blocks are then
stacked to form a hierarchical variational autoencoder [27]. With its design, they achieve impressive
generative capabilities. However, the SetVAE is not fully probabilistic as the reconstruction loss is
approximated with a Chamfer distance. LION, similar to our work, is another variational autoencoder
that maintains a one-to-one mapping throughout the network, enabling probabilistic evaluation [26].
They only implicitly do so by using the l2-loss as their reconstruction loss term. Again, similar to our
work, they also encode their points, but in a higher dimensional space. This unfortunately leads to
information about the shape being stored in here, preventing direct sampling from this space. Finally,
similarly to SetVAE, evaluating the quality of representations in LION, a hierarchical latent variable
model, poses challenges.

One work that succeeded in being optimized fully probabilistically is PointFlow [24]. PointFlow uses
a continuous normalizing flow (CNF) both as prior and decoder, not unlike previous works seen on
images [12, 28]. Intuitively, one CNF can be considered modeling the distribution of shapes, while
the other one is modeling the point distribution given the shape. This resembles VF-Net’s network
architecture in that the encoder maps a global latent space and a point encoding for each point in the
input point cloud. However, PointFlow’s two CNFs are training separately in a two-step process in
contrast to ours. The method is, however, very slow [14]. In our case, PointFlow would have required
200 GPU days of training, and we exclude it from our baselines.

In dentistry, extrapolation of the distal and mesial sides of the tooth is a well-known task. Qiu et al.
[29] presents an impressive attempt to use more classical computational geometry methods, not
relying on training data. They attempt to reconstruct the missing parts of the distal and mesial sides
of the tooth. This leads to a very smooth extrapolation which performs well. Many articles within
dentistry take this a step further, e.g. attempting to extrapolate not just the sides, but also the roots of
the teeth [30–32]. We are optimistic that our model would be able to adapt to such a task given that
dental cone beam computed tomography (CBCT) of the dental roots was available in the training data.
Unfortunately, CBCT scans are incredibly expensive and rare, thus we do not have a large enough
dataset to train a neural network.

5 Experimental results

Limitations. Sampling with VF-Net presents challenges when the sampled shape’s topology differ
from the point encodings. When trained on ShapeNet [33], we can see from Table 4 in the supplemen-
tary materials that VF-Net perform excellent in terms of reconstruction [14, 34]. However, VF-Net
displays its limitations when it comes to to generating new samples. This is due to information of the
Table 1: Chamfer distances (CD) and earth mover’s distances (EMD) are multiplied by 100. Bracket sim and Gap sim are extrapolation performances of the models. PVD was trained for shape completion, while the remaining were evaluated on untrained extrapolation.

<table>
<thead>
<tr>
<th>Method</th>
<th>FDI 16 Tooth</th>
<th>All FDIs</th>
<th>Bracket sim</th>
<th>Gap sim</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CD</td>
<td>EMD</td>
<td>CD</td>
<td>EMD</td>
</tr>
<tr>
<td>DPC</td>
<td>8.51</td>
<td>42.86</td>
<td>5.67</td>
<td>35.8</td>
</tr>
<tr>
<td>SetVAE</td>
<td>21.01</td>
<td>60.13</td>
<td>9.98</td>
<td>51.48</td>
</tr>
<tr>
<td>PVD</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>LION</td>
<td>5.34</td>
<td>22.87</td>
<td>3.02</td>
<td>9.66</td>
</tr>
<tr>
<td>FoldingNet</td>
<td>5.35</td>
<td>34.90</td>
<td>3.43</td>
<td>31.25</td>
</tr>
<tr>
<td>VF-Net</td>
<td><strong>1.07</strong></td>
<td><strong>5.92</strong></td>
<td><strong>0.97</strong></td>
<td><strong>5.30</strong></td>
</tr>
</tbody>
</table>

shape being stored in the latent point encodings, shown in Fig. 3a. The latent point encoding form a non-continuous distribution, posing challenges for sampling new models. Addressing this issue could involve training a flow or diffusion prior on the point encodings, similar to the approach used in LION [26]. However, since this was not the focus of our model, we did not pursue this idea.

**FDI 16 Tooth Data.** Next, we assessed VF-Net’s performance on dental data. The data handling can be found in supplementary section 7.1. We evaluated the reconstruction using Chamfer distances and earth mover’s distance [35],

$$\text{EMD}(X, Y) = \min_{\phi: X \rightarrow Y} \sum_{x \in X} ||x - \phi(x)||_2.$$  \hspace{1cm} (7)

The earth mover’s distance measures the least expensive 1-to-1 transportation between two point sets. However, this is quite computationally expensive and thus rarely used for model optimization [36]. First, we evaluate VF-Net’s performance on our new FDI 16 tooth dataset, and subsequently on a proprietary dataset containing all teeth from which the FDI 16 tooth dataset is a subset.

The FDI 16 dataset contains 7,732 meshes split into train/val/test sets. Training details can be found in supplementary section 7.2 and the reconstruction performances can be found in Table 1. On the FDI 16 data, VF-Net outperforms its peers both when measured using chamfer distances and earth mover’s distances. More reconstruction examples from VF-Net can be found in supplementary Fig. S2. Note that, Point-Voxel Diffusion (PVD) [25] is unable to return the same tooth when embedded, instead it returns a randomly sampled tooth. Therefore, it has been excluded in the comparison.

Figure 3(a): Left: While the airplane from ShapeNet is accurately reconstructed, it poses a challenge in terms of sampling due to its non-continuous distribution in the latent point encoding. Right: An incisor and its corresponding point encodings. Notably, the encoded points correctly reflects the missing sides of the tooth. Sampling and decoding from this region of the latent point encodings enables extrapolation in 3D space.

Figure 3(b): Left: To illustrate the hole reconstruction problem, we present an example where the red points are removed from the point cloud. Right: The latent point encodings remain highly similar to the shape of the encoded points prior to the deletion of points. Sampling the missing area becomes a straightforward task by sampling within the corresponding empty region of the latent point encoding.
Figure 4: FoldingNet facilitates mesh decoding by interconnecting its latent grid. Left: The original input mesh and point cloud are shown with their uneven resolution. Middle: Point cloud and Mesh reconstruction by FoldingNet. Notably, the mesh reconstruction has a gap on the right side of the mesh distorted facets near this border. Right: VF-Net’s mesh and point cloud reconstruction are shown. Our mesh is composed of much more regular facets compared to input point cloud.

**Mesh Reconstruction from Point Cloud.** Fig. 4 shows an example of mesh and point cloud reconstructions from FoldingNet and VF-Net. When visually evaluating the point clouds, both models exhibit a smooth folding behavior. In fact, FoldingNet’s evenly distributed points may even appear to be advantageous. However, upon examining the mesh reconstructions, a different picture emerges. The meshes were constructed by assembling a mesh grid in the latent point encoding as shown in Fig. 2. We decode these assuming a smooth folding, in that points close and connected in the latent space will continue to be so in the output space. In FoldingNet’s mesh reconstruction in Fig. 4, reveals the model deviates notably from the expected folding pattern. Instead, the edge of the grid is observed to reside across the bottom right of the tooth. The region near this area are highly distorted, creating elongated facets that sometimes intersect with one another. Whereas, the VF-Net mesh in Fig. 4 has a more uniform mesh resolution. However, VF-Net’s reconstructions often exhibit excessive smoothness and lack the desired level of detail. A common behavior observed in variational autoencoders [4, 37, 38].

**Variance Estimation for Point Clouds.** The predicted variance from the variance network has been visualized in Fig. 5, where red indicates a higher variance and green indicates a lower variance. Interestingly, the network assigns higher variance to the fifth cusp and to aligner attachments, observed on the two teeth to the left in Fig. 5. Both features are only present in a subset of individuals. The occlusal surface consistently exhibits moderate variance, while the tooth-gingiva border, characterized by frequent directional changes, is associated with the highest variance. The latter is an artifact of the segmentation.

**Table 2:** Sampling performances. MMD have been multiplied by 100.

<table>
<thead>
<tr>
<th>Method</th>
<th>MMD(↓) CD</th>
<th>EMD</th>
<th>COV(↑) CD</th>
<th>EMD</th>
<th>1-NN(↓) CD</th>
<th>EMD</th>
</tr>
</thead>
<tbody>
<tr>
<td>SetVAE</td>
<td>37.78</td>
<td>66.60</td>
<td>11.74</td>
<td>9.91</td>
<td>97.96</td>
<td>97.86</td>
</tr>
<tr>
<td>DPC</td>
<td>36.23</td>
<td>66.13</td>
<td>10.82</td>
<td>10.33</td>
<td>98.74</td>
<td>98.81</td>
</tr>
<tr>
<td>PVD</td>
<td><strong>21.00</strong></td>
<td>52.15</td>
<td><strong>44.34</strong></td>
<td>42.94</td>
<td><strong>63.60</strong></td>
<td>61.77</td>
</tr>
<tr>
<td>LION</td>
<td>21.66</td>
<td>53.32</td>
<td>43.92</td>
<td><strong>43.36</strong></td>
<td>71.54</td>
<td>67.46</td>
</tr>
<tr>
<td>VF-Net (Ours)</td>
<td>21.25</td>
<td><strong>51.98</strong></td>
<td>40.62</td>
<td>25.23</td>
<td>77.47</td>
<td>73.51</td>
</tr>
</tbody>
</table>

**Model Sampling Performances.** To compare sampling performances, we deploy the metrics introduced by Yang et al. [24]. Namely three metrics: Minimum matching distance (MMD) — a metric that measures the average distance to its nearest neighbor point cloud. Coverage (COV)
Figure 5: The predicted relative variance visualization of VF-Net highlights areas of high and low variance, represented by red and green, respectively. Notably, the carabelli cusp and aligner attachment areas exhibit high variance, two features only present in a subset of individual. The highest variances tend to be observed at the mesh border and occlusal surface. These signals align with expectations of where noisy information most likely stem from.

measures the fraction of point clouds in the ground truth test set that is considered the nearest neighbors for the generated sample. 1-nearest neighbor accuracy (1-NNA) uses a 1-NN classifier to classify whether a sample is generated or from the ground truth dataset, 50% meaning they are indistinguishable using this metric.

The sampling performances of the models can be found in Table 2. VF-Net demonstrates superior performance compared to diffusion point cloud (DPC) [34] and SetVAE. Although VF-Net is a competitive model, it falls slightly behind in comparison to PVD and LION [25, 26]. As our primary task is reconstruction and extrapolation, our performances reflect equally. Due to the shape of the latent point encoding and how it may vary (Fig. 3a) sampling grid shapes each time may lead to artifacts which hinder performance on the previously mentioned metrics. Especially the sharp edges on the meshed in Fig. 4 would affect performance. Thus, we trained a minor network equating to one fold of VF-Net to output a point encoding from the latent representation. We emphasize that this is completely unnecessary for regular sampling.

**All FDIs Data set.** We repeated our experiment on a proprietary dataset that include all teeth available, excluding primary teeth (milk teeth). This dataset totals 84,496 teeth close to evenly distributed with respect to FDIs, except for wisdom teeth which are significantly rare. The train-test split was on a jaw level. Training details can found in section 7.4, while the reconstruction performances are in Table 1, where we empirically show that VF-Net also performs well on more complex data. Fig. S3 in the supplementary shows corresponding model samples. We observe that the model can sample from all four major teeth types, incisors, canines, pre-molars, and molars. Furthermore, we followed FoldingNet’s proposed evaluation method of classifying the input point cloud from the latent space. Using a linear support vector machine on this 32-class problem, we achieved a classification accuracy of 85.27% using the latent space of VF-Net. In the same test, FoldingNet achieved an impressive 95.46%. The discrepancy is likely due to Gaussian prior restricting the structure of the latent space. As a sanity check, we retrained a model with a less strict KLD ($\beta=0.01$), where we saw the classification performance increase to 92.34%.

**Simulated Shape Completion.** In dental reconstruction, inferring the obstructed side of a tooth and reconstructing the tooth surface beneath braces’ brackets are key challenges. Paired data of the obstructed and unobstructed surface is exceedingly rare. Therefore, it is valuable to develop a model capable of extrapolating these surfaces without explicit training on such data. As we lack a ground truth for both cases, we instead simulate the extrapolation on the test set. This is done by sampling a point outside the tooth and deleting the 200 nearest neighbors to that point, which equates to under 10% of the point cloud. An example of the synthetic holes is shown in Fig. 3b. From the input point cloud, we removed the red points. The remaining gray points are then encoded by VF-Net, and the resulting reconstruction and its encoded points are displayed on the right side. The latent point encoding shapes remain highly similar and extrapolation is done by simply sampling in the point encoding space. For this evaluation, we sample a higher number of points in the latent encoding and calculate the distance from the red deleted points to their nearest neighbor in the inpainted point cloud. The performance averaged across the test set can be found in Table 1, under "Bracket sim" and "Gap sim", simulating removed bracket and the gap between teeth respectively. Here, VF-Net outperforms its peers when it comes to untrained extrapolation. Note that PVD was trained specifically for these extrapolations, so it is expected to perform better. On the other hand, we were unable to perform shape completion using LION. Shape completion using latent points from the original tooth contained information about the shape and rendered a fair comparison infeasible.
We compared our latent presentation to those of FoldingNet, as it is the comparative model with the most interpretable latent variables. For this task, we attempted to add and remove tooth wear, see Fig. 6. This was done by navigating the latent space of the model in the direction of tooth wear or away from it. The direction of tooth wear was determined by calculating the average change in latent representations when encoding 10 teeth from their counterparts with synthetically induced tooth wear. These teeth were manually sculpted to simulate tooth wear, see supplementary Fig. S4. We observe behavior that closely aligns with our expectations of how the tooth would change when adding or subtracting tooth wear.

Table 3: Percentage teeth which had classification prediction increase according to expectation when moved in the tooth wear direction. L, M, H denotes light, medium, and heavy wear respectively.

<table>
<thead>
<tr>
<th>Method</th>
<th>L → H</th>
<th>L → M</th>
<th>M → L</th>
<th>M → H</th>
<th>H → M</th>
<th>H → L</th>
</tr>
</thead>
<tbody>
<tr>
<td>FoldingNet</td>
<td>91.77</td>
<td>91.77</td>
<td>95.02</td>
<td>94.89</td>
<td>97.80</td>
<td>97.80</td>
</tr>
<tr>
<td>VF-Net (ours)</td>
<td><strong>92.11</strong></td>
<td><strong>99.31</strong></td>
<td><strong>97.04</strong></td>
<td><strong>96.37</strong></td>
<td><strong>98.24</strong></td>
<td><strong>99.12</strong></td>
</tr>
</tbody>
</table>

To quantify the performance, we trained a small PointNet model [11] on a proprietary dataset of 1400 teeth that were annotated with light/medium/heavy tooth wear. Subsequently, we evaluated the model’s ability to correctly classify teeth by determining if a movement in the latent space along the tooth wear direction led to an increase or decrease in the predicted classification. In Table 3, the arrow indicates the direction of the vector used for movement in the latent space. Specifically, "L → H" denotes that we moved from light wear to heavy wear using the corresponding vector obtained from the 10 sculpted teeth as mentioned earlier. The findings presented in Table 3 indicate VF-Net’s latent representations show greater robustness.

6 Conclusion

We have introduced the Variational FoldingNet (VF-Net), a fully probabilistic point cloud model in the same spirit as the original variational autoencoder [4, 5]. The noteworthy technical innovation is the introduction of a per-point encoder network that replaces the commonly used Chamfer distance, which allows us to work with normalizable probability densities. We demonstrate our method in an array of different experiments including reconstruction of our newly released FDI 16 Tooth data. As the point encodings share similar topology as the FDI 16 Tooth data, sampling, and extrapolation becomes remarkably simple. We observe significant improvements over current state-of-the-art models in terms of reconstruction and untrained extrapolation. Additionally, we demonstrate that this performance extends to our proprietary dataset including the remaining teeth from the jaw scans. Finally, we also demonstrate robust representations when it comes to interpolation or modification of the reconstructions.
Acknowledgments and Disclosure of Funding

This work was funded by the Innovation Fund Denmark (1044-00172B) and the Novo Nordisk Foundation through the Center for Basic Machine Learning Research in Life Science (NNF20OC0062606). It also received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research, innovation programme (757360). Søren Hauberg was supported in part by a research grant (42062) from VILLUM FONDEN.

References


7 Supplementary Material

Figure S1: In the above plot, we observe that the euclidean distance acts as an upper bound for the chamfer distance. In the majority of cases, when optimized using VF-Net, these distances are identical. This empirical observation provides support for our claim that the Chamfer distance can be effectively substituted with an appropriate encoder choice.

Table 4: Both Chamfer distances (CD) and earth mover’s distances (EMD) are multiplied by 1000, and for both, lower values indicate better reconstruction performance.

<table>
<thead>
<tr>
<th>Method</th>
<th>ShapeNet Airplanes</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CD</td>
<td>EMD</td>
</tr>
<tr>
<td>DPC</td>
<td>0.18</td>
<td>47.82</td>
</tr>
<tr>
<td>SetVAE</td>
<td>0.14</td>
<td>30.60</td>
</tr>
<tr>
<td>PVD</td>
<td>0.31</td>
<td>90.45</td>
</tr>
<tr>
<td>LION</td>
<td>0.061</td>
<td>10.19</td>
</tr>
<tr>
<td>FoldingNet</td>
<td>0.079</td>
<td>31.47</td>
</tr>
<tr>
<td>VF-Net (ours)</td>
<td>0.031</td>
<td>7.31</td>
</tr>
</tbody>
</table>

7.1 Data Handling

For the dental scan experiments, the point clouds used were constructed from vertices and facet midpoints. The cardinality of the raw point clouds varied significantly, ranging from around 2,000 to 65,000 points. To handle this variation, we subsampled 2048 points from each point cloud during each epoch. Selecting an appropriate normalization method is crucial for FoldingNet and VF-Net, as it has a significant impact on the extent of deformation needed for the initial points to reach the desired final output. To allow for the point encodings to subside in the planar patch $[-1, 1]^2$. Thus, we scale the data so 99.5% of all points are within the unit sphere. During training, data augmentation was applied for half the point clouds. This involves flipping the $x$-axis 35.3% of the time, randomly scale the point cloud between [80%, 120%] of its size, randomly slightly tilt the data away from the $y$-axis up to 15°, and finally add slight noise to the input points.
7.2 FDI 16 Training Details

The network was trained using the adamax optimizer with an initial learning rate of 0.001. A batch size of 64 was used, and the training process was conducted for 10000 epochs. We employed a KLD warm-up [27] during the initial 2500 epochs, whenever applicable. VF-Net’s initial training a constant variance was used. Subsequently, a separate training of 100 epochs to train only the variance network [39].

7.3 FDI 16 Reconstructions

![Figure S2: Examples of reconstructions of meshes from the FDI 16 dataset.](image)

7.4 All FDI Training Details

For the experiments on the proprietary dataset with all teeth includes, every model shares the identical architecture and size as used for the FDI 16 experiment. However, the training is limited to 1250 epochs, with a KLD warm-up phase comprising one-fourth of the total epochs, when applicable. Again, a separate training run of 100 epochs was done to tune the variance network [39].

7.5 Sampled All FDI Teeth

![Figure S3: A showcase of meshes sampled by VF-Net. All four major modalities are covered: Incisor, canine, pre-molar, and molar, the four major types of teeth.](image)
7.6 Synthetic Toothwear Teeth

Figure S4: Two of the ten manually sculpted teeth to simulate tooth wear. **Left**: Highlighted in red are areas that have higher values compared to the original reconstruction. **Middle**: The original reconstruction. **Right**: Areas depicted in blue are lower than the original mesh.