Федеральное государственное автономное образовательное учреждение
высшего образования «Национальный исследовательский университет
«Высшая школа экономики»

Факультет компьютерных наук
Основная образовательная программа
Прикладная математика и информатика

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Исследовательский проект на тему "Методы кросс-модального предобучения моделей для извлечения локальных представлений 3D данных без учителя"

Выполнил студент группы 182, 4 курса,
Чураков Игорь Александрович

Руководитель КР:
научный сотрудник, Артемов Алексей Валерьевич

Москва 2022

Содержание

1 Аннотация

Обучение без учителя это широко используемый ряд методов, позволяющий получать полезные сжатые представления сложных структур данных. При работе с трехмерными данными важно учитывать, что одни и те же объекты могут быть представлены разными модальностями. В этой работе мы исследуем, как можно переходить от представлений в одной модальности к представлениям в другой.

2 Abstract

Unsupervised learning is a widely used technique that allows obtaining useful representations of complex data structures. When dealing with 3D data, it is crucial to consider the fact that shapes can be represented by different modalities. In this paper we study how one can translate shape representations between different modalities.

3 Ключевые слова

Извлечение признаков, 3D модели, векторные представления, методы обучения без учителя, кроссмодальное обучение

4 Введение

4.1 Описание предметной области

Появление больших неразмеченных коллекций данных привело к значительному росту deep learning методов обучения без учителя для 2D компьютерного зрения. Модели обучаются на большой выборке без разметки и выучивают эмбеддинги, которые в дальнейшем можно использовать для решения последующих задач. Многие из этих методов естественным образом обобщаются на 3D компьютерное зрение. Отсутствие больших качественных размеченных датасетов и высокая дороговизна аннотирования делает проблему предобучение в 3D CV очень актуальной. Кроме того, отличие 3D компьютерного зрения от 2D заключается в том, что одни и те же данные в первом можно представлять по разному, то есть используя различные модальности (3D облака точек, меши, восксельные сетки, sdf, CAD модели). Эти представления не равнозначны: так например из CAD моделей можно получить любые другие формы данных, а из облаков точек проблематично восстановить остальные форматы. Из-за этого возникает проблема выбора модели и данных для предобучения и инференса. В данной работе будут рассмотрены следующие модальности: облака точек, полигональные модели.

Облако точек представляет из себя множество вершин пространстве и задается набором трехмерных координат. Это самый популярный в 3D компьютерном зрении формат данных. Облако точек можно получить сэмплируя точки на поверхности объектов, представленных модальностями с более точной геометрией (полигональные модели, CAD модели) или отсканировав реальный объект трехмерным сканером. Сложность в работе с облаками точек заключается в том, что они довольно грубо передают геометрию объекта и представляют из себя данные нерегулярной природы: наборы координат, отличающиеся только порядком точек, представляют одну и ту же фигуру.

Полигональная модель (или меш) это набер вершин, ребер и граней. Она задается набором вершин (облаком точек) и индексами вершин которые об-

разуют грани. Это самый популярный формат представления трехмерных объектов в компьютерной графике. Меши более экспрессивны чем облака точек и позволяют передать геометрию объекта с практически любой точностью. Ввиду особенностей представления, этот формат данных такой же нерегулярный как облака точек.

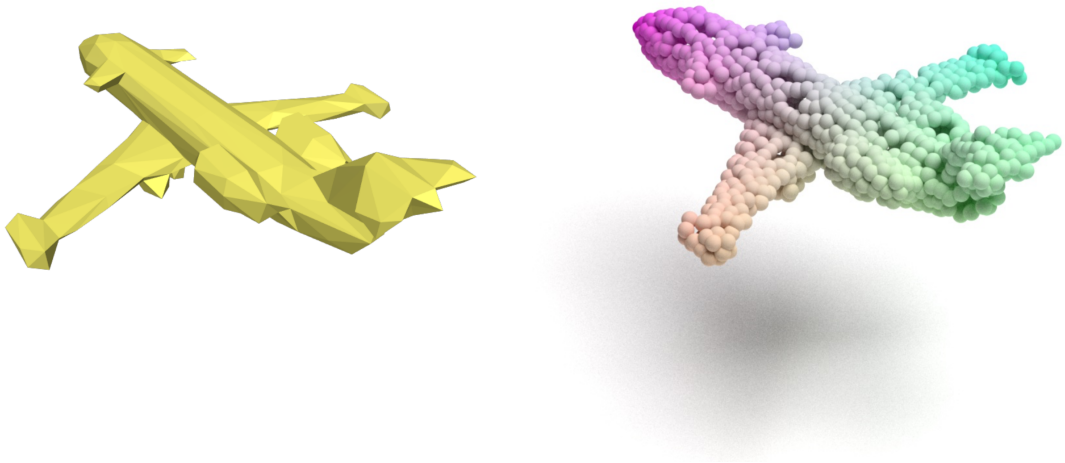

Рис. 1: Модель представленная мешем (слева) и облаком точек (справа)

Разные представления лучше подходят для определенных академических или практических применений. Облака точек являются самыми исследованными, нейросети работающие с ними показывают лучшие результаты в таких задачах как сегментация и классификация. Работ, которые исследуют нейросети, напрямую работающие с полигональными моделями, гораздо меньше, а модели для мешей, несмотря на то, что меши более точно передают детали объектов, учат менее информативные представления и сильно зависят от качества геометрии. В связи с этим естественным кажется исследовать различные способы объединения модальностей. В ходе исследования мы попытались совместно (кроссмодально) обучать нейросети для мешей и облаков точек, но остановились на проецировании уже обученных эмбеддингов из одного латентного пространства (пространства мешей) в другое (пространство облаков точек).

4.2 Постановка задач

В данной работе была поставлена задача разработки и исследования методов построения локальных векторных представлений для 3D объектов с использованием информации из нескольких модальностей (мешей и облаков точек). Подзадачи исследовательского проекта:

1 Разработать метод предобучения для 3D компьютерного зрения, учитывающий локальнаые и глобальные признаки объектов.

2 Обучить модели, извлекающие локальные признаковые описания для фигур, представленных модальностями мешей и облаков точек. Протестировать полученные модели.

3 Придумать способ переходить из пространства эмбеддингов одной модальности, в пространство эмбеддингов другой, протестировать полученный метод.

5 Обзор литературы

5.1 Используемые модели

5.1.1 MeshNet [1]

MeshNet работает напрямую с полигональными моделями, на вход модель получает меш, на выходе выдает признаковые описания для каждой грани. Модель устроена следующим образом: для каждой грани извлекаются координаты центра, вектор нормали, вектора соединяющие центр с углами грани, индексы соседних граней. Эти данные обрабатываются пространственными и структурными дескрипторами, а затем пропускаются через последовательность определенных авторами блоков меш сверток.

5.1.2 DGCNN [2]

DGCNN работает с облаками точек, на вход модель получает облако, на выходе получаются признаковые описания для всех точек. На каждом шаге алгоритма по облаку строится граф ближайших соседей, близость точек определяется близостью представлений точек в латентном пространстве (в первом приближении – координатами точек). После построения графа, он обрабатывается с помощью серии графовых сверток.

5.2 Используемые наборы данных

5.2.1 ABC Dataset [3]

Набор из миллиона CAD моделей. Все данные взяты с ресурса Onshape - программное обеспечение для автоматического проектирования, которое предоставляет доступ к большой коллекции CAD моделей. В данных нет разметки для классификации или сегментации.

5.2.2 ModelNet40 [4]

Набор из 12311 полигональных мешей, 40 различных категорий. Мы использовали версию Manifold40, которую рекомендуют использовать авторы MeshNet. В ней модели упрощены и у них исправлена геометрия.

5.2.3 COSEG [5]

Датасет состоит из 8 малых и 4 больших наборов фигур, разделенных по категориям. Для каждого из набора, объекты представлены мешами, в COSEG есть разметка для сегментации.

5.3 Методы обучения

5.3.1 Contrastive Learning

В 2D компьютерном зрении очень популярен класс методов Contrasive Learning для обучения векторных представлений объектов (или частей объектов) без учителя. Процесс обучения можно описать так: Пусть есть батч размера N. Модели на вход подаются $2N$ объектов, по две версии на каждый, полученные с помощью аугментаций исходных объектов. Модель выдает $2N$ векторных представлений. Задача contrastive prediction формулируется следующим образом: Для каждого представления у нас есть позитивный пример (эмбеддинг своей аугментации) и $2N-1$ негативных, нужно максимизировать близость между позитивными парами и минимизировать между негативными. Пусть близость между двумя объектами определяется как:

$$sim(a, b) = \frac{a^T b}{\|a\|\|b\|}$$

Тогда функция потерь для каждой позитивной пары (i, j) выглядит так:

$$l(i, j) = -log\frac{\exp(sim(x_i, x_j)/\tau)}{\sum_{k=1}^{2N}[k \neq i]\exp(sim(x_i, x_k)/\tau)}$$

Параметр τ отвечает за температуру. Итоговая функция потерь это сумма $l(i, j)$ и $l(j, i)$ для всех позитивных пар i, j в батче.

Самые известные статьи, использующие подобный подход это MoCo [6] и SimCLR [7].

Изначально все Contrastive Learning методы выучивали глобальные эмбеддинги, то есть представления на уровне целого объекта, а значит не очень подходили для обучения локальных эмбеддингов. Однако позднее появилось несколько работ, которые исследуют методы контрастного предобучения для обучения локальных представлений. В [8] между собой контрастируются выделенные из изображений патчи, а в [9] отдельные пиксели, мы вдохновля-

лись этими работами при разработке нашего фреймворка.

5.3.2 Кроссмодальное обучение

Несколько подходов для кроссмодального 3D машинного обучения были предложены ранее. В статье [10] авторы используют разные модальности как аугментации объектов для Contrastive Learning. В качестве модальностей они выбрали меши, облака точек и рендеры трехмерных объектов. Дополнительно они формулируют contrastive prediction отдельно для рендеров, в качестве положительных пар используются изображения одно и того же объекта под разными углами. Похожие идеи исследуются в [11], там обучается модель для облаков точек и рендеров. Кроссмодальная часть состоит в том что эммбеддинги фигур из разных модальностей склеиваются и модель решает задачу бинарной классификации, пытаясь определить правда ли, что оба эмбеддинга пришли из одной модальности. Главный недостаток обоих подходов заключается в том, что они учат глобальные представления, на уровне целых объектов. В [12] учится VAE для каждой из трех модальностей: облака точек, меши и рендеры фигур. Каждый объект для каждой модальности восстанавливается из всех трех латентных представлений. Помимо этого авторы используют разметку, дополнительно обучая модель решать задачу классфикации, поэтому метод нельзя отнести к классу обучающихся без учителя. Также из работ по 3D CV, работающих с разными модальностями, стоит упомянуть [13]. Авторы работают с RGB-D сканами (изображение плюс информация о глубине) и с помощью моделей для работы с изображениями и с воксельными сетками извлекают совместные представления.

6 Предложенный метод

В качестве метода предобучения использовался contrastive prediction, сформулированный в SimCLR [7], адаптированный для доменов облаков точек и мешей. Основной упор в работе был сделан на выучивание локальных пред-

ставлений, поэтому также как в [8] и [9] было добавлено предобучение на уровне локальных частей трехмерных объектов.

6.1 Предобучение на полигональных моделях

Фреймворк для предобучения на мешах устроен следующим образом: на вход модели подается два батча размера N, фигуры в них отличаются поворотом на случайный угол от $-45°$ до $45°$ по каждой из трех осей и добавлением случайного шума к центрам граней. Модель извлекает признаки для каджой грани, для каждого меша в батчах. Затем решаются две contrastive prediction задачи: глобальная и локальная.

Глобальная задача состоит в том, что между собоой контрастируются представления целых объектов, полученные усреднением эмбеддингов каждой грани. По аналогии с 2D CV, где контрастируются аугментированные изображения.

Локальная задача заключается в том, что между собой контрастируются эмбеддинги граней аугментированных мешей. Если проводить аналогию с Contrastive Learning в 2D CV, между собой контрастируются соответствующие пиксели в аугментированных картинках.

6.2 Предобучение на облаках точек

Фреймворк для предобучения на облаках точек: на вход модели подается два батча размера N, облака в них отличаются поворотом на случайный угол от $-45°$ до $45°$ по каждой из трех осей и добавлением случайного шума к координатам. Модель извлекает признаки для каджой точки, для каждого облака. Затем решаются две contrastive prediction задачи: глобальная и локальная.

Глобальная задача состоит в том, что между собоой контрастируются представления целых объектов, полученные усреднением эмбеддингов каждой точки.

Для локальной задачи, мы использовали контрастирование между собой частей облака. Для каждой точки известно, на какой грани соответствующего меша она была насэмплирована, для каждой грани, мы усредняли эмбеддинги точек, которые попадают на нее. Получив таким образом естественное разбиение облака точек на патчи, точно так же как для мешей, мы контрастировали между собой эмбеддинги патчей для двух версий одного облака точек. Интуиция за выбором такого метода следующая: мы хотим, чтобы близкие друг к другу точки имели близкие в латентном пространстве эмбеддинги, кроме того, так мы неявно форсируем кроссмодальность, строя на уровне обучения соответствие между облаками точек и мешами.

6.3 Кроссмодальное предобучение

Изначально планировалось придумать фреймворк, позволяющий одновременно оптимизировать сеть для облаков точек и для мешей, строя при это соответствие для эмбеддингов граней и эмбеддингов точек, лежащих на этой грани.

Опробованный нами способ предобучения был устроен так: на каждом шаге метода на вход моделям подавались N фигур, представленные в модальностях облаков точек и мешей. Меши и облака точек аугментировались. Модель для мешей решала свои глобальную и локальную задачи, модель для точек решала свои. При этом в функцию потерь добавлялись две кроссмодальные компоненты: локальная и глобальная.

Глобальная кроссмодальная задача состояла в том, что между собой контрастировались глобальные эмбеддинги мешей и облаков точек в батче. Так, позитивная пара в каждом случае это один и тот же трехмерный объект, представленный в одном случае облаком точек, а в другом мешем. Интуитивно кажется, что это очень хорошая аугментация для Contrastive Learning, подхода, построенного на аугментировании объектов.

При решении локальной задачи для облака точек, оно разбивалось на

патчи, отвечающие граням соответствующего меша. Таким образом у нас получается соответствие локальных эмбеддингов облака точек и меша, представляющих одну трехмерную фигуру. В ходе решения локальной задачи эти патчи контрастировались между собой.

Проблема данного подхода заключалась в том, что кроссомодальная часть функции потерь не оптимизировалась, тогда как локальные и глобальные функции потерь обеих модальностей по отдельности оптимизировались. Мы попробовали внести некоторые модификации в предложенный нами пайплайн, но ни одна из них не помогла. Были опробованы следующие способы: скалирование различных компонент функции потерь, предобучение обеих модальностей по отдельности и последующее обучение кроссмодальной компоненты. При предобучении использовались куски мешей из ABC Dataset и насэмплированные на них облака точек. Вырезанные патчи были довольно простыми и имели хорошую геометрию, поэтому их обработка не должна была вызвать проблемы у сети для мешей. Мы связываем эти неудачи как со сложностью самой задачи, так и с тем, что нейросети, работающие с полигональными моделями менее изучены и извлекают менее информативные признаки по сравнению с нейросетями для облаков точек.

В последствии от совместной оптимизации было решено отказаться и было принято решение изучать возможности проецирования эмбеддингов из домена мешей в домен облаков точек.

6.4 Проецирование эмбеддингов

Как было упомянуто во введении, модели для облаков точек сильнее моделей для мешей, поэтому после неудачи с кроссмодальным обучением мы остановились на выучивании проекции из пространства эмбеддингов граней мешей в пространство эмбеддингов точек.

Модель для точек и для мешей по отдельности предобучаются только на своей модальности. На уровне датасета у нас есть соответствие между

гранями и точками в фигурах, поэтому можно предсказывать эмбеддинги точек, лежащих на текущей грани, по эмбеддингу грани объекта.

Метод устроен следующим образом: меши и облака точек аугментируются также, как при предобучении, затем извлекаются локальные эмбеддинги для каждой точки и каждой грани, каждая грань проецируется с помощью модели-проектора в пространство эмбеддингов облаков точек, считается функция потерь между проецированным эмбеддингом и эмбеддингами точек, которые лежат на данной грани в меше.

Так как на одной грани меша скорее всего будут лежать несколько точек из облака, в качестве одной из модификации проектора было предложено конкатенировать эмбеддинг текущей грани с координатами точки, эмбеддинг которой мы хотим получить в результате проецирования.

7 Эксперименты

7.1 Данные

В качетсве основного датасета для предобучения использовался Manifold40, это версия ModelNet40 с упрощенной и исправленной геометрией. На каждом меше было насэмплировано 1024 точки с помощью Poisson disk сэмплинга [14].

Для тестирования сегментации был использован датасет COSEG. Мы использовали все три больших набора фигур оттуда: Tele-aliens, Vases, Chairs.

7.2 Процедура обучения

Для предобучения облаков точек и мешей модели обучались 100 эпох. Оптимизатор AdamW, параметр weight decay равен $1e^{-5}$, learning rate возрастал с 0 до 0.001 в течение первых четырех эпох. Использованные аугментации: для облаков точек поворот на случайный угол от $-45°$ до $45°$ по каждой из трех осей и добавление случайного шума к координатам точек; для мешей: поворот на случайный угол от $-45°$ до $45°$ по каждой из трех осей и добав-

ление случайного шума к координатам центров граней. Локальная функция потерь комбинировалась с глобальной с разными весами: $0.01, 0.05, 0.1, 0.2$, в ходе обучения выяснилось, что если ее не скалировать, то глобальная компонента не обучается.

Проектор обучался 100 эпох, использовался оптимизатор AdamW, параметр weight decay равен $1e^{-5}$, learning rate равен 0.001 с CosineAnnealing расписанием. При обучении проектора использовались MSE и L1 функции потерь. Используемые аугментации аналогичны тем, что использовались при предобучении.

7.3 Классификация на ModelNet40

7.3.1 Тестирование предобученных моделей

Для тестирования полученных представлений в задаче сегментаций использовался датасет ModelNet40. Целью этого эксперимента было проверить насколько информативные эмбеддинги получаются в результате выбранной процедуры предобучения. Для тестирования классификации извлеченные представления для граней и точек усреднялись по всем граням и точкам для каждой фигуры, а затем на полученных представлениях обучался линейный SVM классификатор.

Модель	**Вес локальной компоненты**	**ModelNet40**
MeshNet	0	0.487
DGCNN	0	0.705
MeshNet	0.01	0.544
DGCNN	0.05	0.824
MeshNet	0.05	0.589
DGCNN	0.1	**0.852**
MeshNet	0.1	0.575
MeshNet	0.2	0.5859
3D-GAN (2016) [15]	–	0.833
FoldingNet (2018) [16]	–	0.884
Jigsaw3D (2019) [17]	–	0.906
Rotation3D (2020) [18]	–	0.907
ParAE (2021) [19]	–	0.916
POS-BERT (2022) [20]	–	**0.921**

Таблица 7.1: Сравнение качества классификации. Метрика: accuracy

В таблице 7.1 можно наблюдать результаты в задаче классификации. Хочется отметить, что добавление локальной компоненты в функцию потерь дает значительный прирост качества классификации, что неочевидно, потому что при решении локальной задачи мы контрастируем эмбеддинги точек или граней только в рамках одной фигуры. Получается, что обучение на уровне локальных признаков очень важно даже для решения задач на уровне всего объекта. Также можно заметить, что увеличение веса локальной компоненты функции потерь положительно влияет на информативность полученных эмбеддингов, но для MeshNet при увеличении с 0.05 до 0.1 прироста качества в отличие от DGCNN уже не происходит. В данной таблице отлично проиллюстрировано, насколько более информативные представления извлекает модель, работающая с облаками точек по сравнению с моделью, работающей с мешами.

В нижней части таблицы приведено сравнение с другими методами предобучения для трехмерных объектов, для каждого года выбрана модель, показывающая лучший результат в такой же постановке: предобучение модели и обучение линейного SVM классификатора на извлеченных признаках объектов. Все приведенные модели, кроме 3D-GAN работают с облаками точек, 3D-GAN работает с воксельными сетками. Результаты взяты из статьи POS-BERT [20]. Стоит отметить, что все эти модели предобучались на гораздо более большом и разнообразном наборе данных ShapeNet [21] и использовали облака точек большего разрешения, мы не использовали ShapeNet ввиду низкого качества геометрии представленных в нем мешей, потому что основная цель данной работы это исследование связей между обучением на обеих модальностях. Подводя итог, можно сказать, что модели выучивают информативные эмбеддинги, а методы контрастного предобучения из 2D CV успешно адаптируются под 3D CV.

7.3.2 Тестирование проектора эмбеддингов

В рамках этого эксперимента было проверено, как ведут себя эмбеддинги граней меша после проецирования в латентное пространство эмбеддингов точек в задаче классификации.

Функция потерь	локальный вес	Метод	ModelNet40	ModelNet40 до проекции	DGCNN ModelNet40
MSE	0	без координат	0.449	0.487	0.705
L1	0	без координат	0.464	0.487	0.705
MSE	0	с координатами	0.644	0.487	0.705
L1	0	с координатами	0.643	0.487	0.705
MSE	0.05	с координатами	0.734	0.589	0.824
L1	0.05	с координатами	0.774	0.589	0.824
MSE	0.1	с координатами	0.752	0.575	0.852
L1	0.1	с координатами	0.777	0.575	0.852
MSE	0.05	глобальный эмбеддинг	0.522	0.589	0.824
L1	0.05	глобальный эмбеддинг	0.530	0.589	0.824

Таблица 7.2: Сравнение качества классификации на проецированных эмбеддингах. Метрика: accuracy

В таблице 7.2 можно наблюдать сравнения качества классификации для эмбеддингов, спроецированных из латентного пространства признаков граней, в латентное пространство признаков точек. Обозначения в колонке ме-

тод: без координат – эмбеддинг каждой грани проецируется как есть; с координатами – эмбеддинг каждой грани конкатенируется с каждой из точек, которые на нем лежат, проецируется полученный вектор; глобальный эмбеддинг – проецируются только глобальные эмбеддинги мешей в глобальные эмбеддинги облаков точек, глобальные эмбеддинги получаются усреднением всех локальных в рамках одного объекта.

Стоит отметить, что конкатенация эмбеддинга грани с точкой перед проецированием дает существенный прирост в качестве классификации, если убрать этот компонент, то качество классификации после проецирования оказывается хуже, чем качество до. Такое же поведение можно заметить если попытаться проецировать только глобальный эмбеддинг. Можно сделать вывод, что соответствие на уровне локальных признаков, даже для решения глобальных задач, при проецировании эмбеддингов также важно, как при предобучении. При добавлении координат точек, эмбеддинги которых мы пытаемся предсказать, качество классификации улучшается и превосходит результат до проекции. Этот эффект усиливается с увеличением веса локальной компоненты функции потерь при предобучении. И, хоть до результатов DGCNN модели для мешей все равно далеко, при проецировании удается значительно улучшить метрику, в лучшем случае удалось добиться прироста в целых 20 пунктов. Относительно функции потерь, можно сказать, что L1 в целом лучше подходит для данной задачи. Подводя итоги: проектор эмбеддингов это мощный инструмент для улучшения качества классификации на мешах.

7.4 Сегментация на COSEG

Обученные модели были протестированы на задаче сегментации на наборе данных COSEG. В качестве сценария для тестирования был выбран few-shot learning. В рамках каждого эксперимента предобученная модель 4 раза дообучалась на случайно выбранном помножестве данных в размере 5% от всей

выборки и тестировалась на оставшейся части данных. Итоговый результат представлял из себя усреднение по всем четырем запускам.

7.4.1 Тестирование предобученных моделей

Модель	**локальный вес**	**Aliens**	**Vases**	**Chairs**
MeshNet	0	0.384	0.379	0.717
DGCNN	0	0.441	0.502	0.728
MeshNet	0.05	0.597	0.586	0.776
DGCNN	0.05	0.598	0.597	0.879
MeshNet	0.1	0.659	0.543	0.744
DGCNN	0.1	0.63	0.611	0.883

Таблица 7.3: Сравнение качества сегментации. Метрика: mIoU

В таблице 7.3 приведено сравнение результатов в задаче сегментации, можно заметить, что в этой задаче разрыв между облаками точек и мешами не такой большой, а меши иногда даже выигрывают по качеству. Опять же можно пронаблюдать, что с ростом веса локальной компоненты функции потерь улучшается метрика качества, причем в отличие от классификации, в сегментации больший прирост качества с ростом множителя у MeshNet.

7.4.2 Тестирование проектора эмбеддингов

Тестирование проектора эмбеддингов происходило следующим образом: модель для мешей извлекала признаковые описания для каждой грани, затем эмбеддинг грани конкатенировался с координатами центра грани и проецировался, таким образом предсказывался эмбеддинг точки, которая должна была попасть ровно в центр текущей грани, а значит иметь такую же сегментационную разметку.

Модель	функция потерь	локальный вес	Aliens до	Aliens	Vases до	Vases	Chairs до	Chairs
MeshNet	MSE	0	0.384	0.376	0.379	0.442	0.717	0.68
MeshNet	L1	0	0.384	0.377	0.379	0.424	0.717	0.668
MeshNet	MSE	0.05	0.597	0.595	0.586	0.564	0.776	0.77
MeshNet	L1	0.05	0.597	0.603	0.586	0.57	0.776	0.762
MeshNet	MSE	0.1	0.659	0.643	0.543	0.552	0.744	0.742
MeshNet	L1	0.1	0.659	0.64	0.543	0.523	0.744	0.754

Таблица 7.4: Сравнение качества сегментации при проецировании эмбеддингов граней. Метрика: mIoU

В таблице 7.4 можно наблюдать результаты эксперимента. Во всех случаях, кроме одного, качество при проецировании незначительно ухудшилось, причем чем больше вес локальной компоненты, тем меньше деградация качества при трансляции, таким образом, мы не получили выигрыш в качестве для MeshNet, зато показали, что почти без потерь информативности можем переходить от эмбеддингов фейсов, к эмбеддингам произвольных точек на их поверхности. Также в рамках данного эксперимента нет разницы какую функцию потерь использовать при обучении проектора.

7.4.3 Выводы

По результатам тестирования проектора эмбеддингов на задачах классификации и сегментации можно сделать выводы, что у нас действительно получилось выучить преобразование, которое по эмбеддингу грани и координатам точки на ее поверхности предсказывает эмбеддинг этой точки в другом латентном пространстве. Данное преобразование значительно повышает качество классификации полигональных моделей, а при больших значениях веса локальной компоненты функции потерь в предобучении не ухудшает качество сегментации.

Что касается выбора функции потерь для обучения проектора, тестирование на COSEG показало, что задача сегментации не чувствительна к выбору функции потерь. В то время как при тестировании на ModelNet40 лучшей оказалась L1 функция потерь, поэтому можно сделать вывод что при обучении нужно использовать ее.

8 Заключение

В проекте было проведено исследование применимости методов Contrasting Learning из 2D CV для трехмерных объектов. Предложенный способ предобучения работает для обеих модальностей и выучивает информативные эмбеддинги локальных частей фигур. В процессе исследования было обнаружено, что добавление contrastive prediction задачи на эмбеддинги в рамках только одной фигуры все равно значительно повышает качество классификации.

Были выучены несколько преобразований, позволяющих по эмбеддингам граней получать эмбеддинги произвольных точек на поверхности меша. С помощью данного преобразования, пользуясь большей выразительностью латентных представлений точек, можно значительно улучшить качество классификации. Потенциальной сферой применения могут быть задачи, в которых нужно искать соотвествие между частями меша и частями облака точек.

Код проекта: https://github.com/iachurakov/dense3dfeatures

Список литературы

[1] Yutong Feng и др. *MeshNet: Mesh Neural Network for 3D Shape Representation*. 2018. arXiv: 1811.11424 [cs.CV].

[2] Yue Wang и др. "Dynamic Graph CNN for Learning on Point Clouds". В: *ACM Trans. Graph.* 38.5 (окт. 2019). DOI: 10.1145/3326362.

[3] Sebastian Koch и др. "ABC: A Big CAD Model Dataset For Geometric Deep Learning". В: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Июнь 2019.

[4] Zhirong Wu и др. "3D ShapeNets: A deep representation for volumetric shapes". В: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, с. 1912—1920. DOI: 10.1109/CVPR.2015.7298801.

[5] Yunhai Wang и др. "Active Co-Analysis of a Set of Shapes". В: *ACM Trans. Graph.* 31.6 (нояб. 2012). ISSN: 0730-0301. DOI: 10.1145/2366145.2366184.

[6] Kaiming He и др. *Momentum Contrast for Unsupervised Visual Representation Learning*. 2019. DOI: 10.48550/ARXIV.1911.05722.

[7] Ting Chen и др. *A Simple Framework for Contrastive Learning of Visual Representations*. 2020. DOI: 10.48550/ARXIV.2002.05709.

[8] Olivier J. Hénaff и др. *Efficient Visual Pretraining with Contrastive Detection*. 2021. DOI: 10.48550/ARXIV.2103.10957.

[9] Pedro O. Pinheiro и др. *Unsupervised Learning of Dense Visual Representations*. 2020. DOI: 10.48550/ARXIV.2011.05499.

[10] Longlong Jing и др. *Self-supervised Modal and View Invariant Feature Learning*. 2020. DOI: 10.48550/ARXIV.2005.14169.

[11] Longlong Jing и др. *Self-supervised Feature Learning by Cross-modality and Cross-view Correspondences*. 2020. DOI: 10.48550/ARXIV.2004.05749.

[12] Sanjeev Muralikrishnan и др. “Shape Unicode: A Unified Shape Representation”. В: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Июнь 2019.

[13] Yunze Liu и др. *P4Contrast: Contrastive Learning with Pairs of Point-Pixel Pairs for RGB-D Scene Understanding*. 2020. DOI: 10.48550/ARXIV.2012.13089.

[14] John Bowers и др. “Parallel Poisson Disk Sampling with Spectrum Analysis on Surfaces”. В: *ACM Trans. Graph.* 29.6 (дек. 2010). ISSN: 0730-0301. DOI: 10.1145/1882261.1866188.

[15] Jiajun Wu и др. “Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling”. В: *Advances in Neural Information Processing Systems*. 2016, с. 82—90.

[16] Y. Yang и др. “FoldingNet: Point Cloud Auto-Encoder via Deep Grid Deformation”. В: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, с. 206—215. DOI: 10.1109/CVPR.2018.00029.

[17] Jonathan Sauder и Bjarne Sievers. “Self-Supervised Deep Learning on Point Clouds by Reconstructing Space”. В: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2019.

[18] Omid Poursaeed и др. “Self-Supervised Learning of Point Clouds via Orientation Estimation”. В: *2020 International Conference on 3D Vision (3DV)*. 2020, с. 1018—1028. DOI: 10.1109/3DV50981.2020.00112.

[19] Benjamin Eckart и др. “Self-Supervised Learning on 3D Point Clouds by Learning Discrete Generative Models”. В: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, с. 8244—8253. DOI: 10.1109/CVPR46437.2021.00815.

[20] Kexue Fu и др. *POS-BERT: Point Cloud One-Stage BERT Pre-Training*. 2022. DOI: 10.48550/ARXIV.2204.00989.

[21] Angel X. Chang и др. *ShapeNet: An Information-Rich 3D Model Repository.* Тех. отч. arXiv:1512.03012 [cs.GR]. Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015.