

УДК 519.6

DOI: [10.26102/2310-6018/2022.36.1.013](https://doi.org/10.26102/2310-6018/2022.36.1.013)

## Нейросетевое устранение шума в полигональных сетках

О.М. Ротова, Н.В. Пивоварова✉

*Московский государственный технический университет имени Н.Э. Баумана,  
Москва, Российская Федерация  
[pivovarova.natasha2013@yandex.ru](mailto:pivovarova.natasha2013@yandex.ru) ✉*

**Резюме.** Одной из актуальных проблем при создании 3D-моделей с помощью систем трехмерного сканирования является автоматическая обработка для устранения шума, полученного из-за использования сканирующих устройств недостаточной точности. Целью исследования является разработка полностью автоматического подхода для решения задачи устранения в полигональных сетках, полученных после трехмерного сканирования. Ведущим методом к решению данной проблемы является использование нейронных сетей, позволяющих произвести все процессы по обработке полигональных сеток автоматически. В статье представлены обзор и сравнительный анализ существующих методов устранения шума в полигональных сетках. Сформулирована математическая постановка задачи устранения шума. Приведено описание используемых алгоритмов для подготовки данных для обучения нейронной сети. Используются метод фильтрации полигональных сеток с помощью билатерального фильтра, метод главных компонент для понижения размерности данных, алгоритм кластеризации k-средних, алгоритм обновления позиций вершин по обновленным нормальным векторам граней. Описываются детали реализации полносвязной нейронной сети прямого распространения. Представляются результаты исследования. Проводится анализ полученных результатов с помощью метрик качества расстояния Хаусдорфа и среднего значения угла между нормальными векторами полигональных сеток с шумом и без шума.

**Ключевые слова:** нейронные сети, наука о данных, полигональные сетки, устранение шума, трехмерное сканирование, билатеральный фильтр.

**Для цитирования:** Ротова О.М., Пивоварова Н.В. Нейросетевое устранение шума в полигональных сетках. *Моделирование, оптимизация и информационные технологии.* 2022;10(1). Доступно по: <https://moitvvt.ru/ru/journal/pdf?id=1093> DOI: 10.26102/2310-6018/2022.36.1.013

## Neural network denoising in polygon meshes

О.М. Rotova, N.V. Pivovarova✉

*Bauman Moscow State Technical University,  
Moscow, Russian Federation  
[pivovarova.natasha2013@yandex.ru](mailto:pivovarova.natasha2013@yandex.ru) ✉*

**Abstract:** One of the most important problems of creating 3D models with the aid of three-dimensional scanning systems is automatic processing to eliminate noise obtained due to the application of scanning devices with insufficient accuracy. The aim of the study is to develop a fully automatic approach for solving the problem of denoising in polygon meshes acquired after three-dimensional scanning. The principal method to overcome this is the application of neural networks that allow processing of polygon meshes to be performed automatically. The article presents an overview and comparative analysis of existing methods of denoising in polygon meshes. The mathematical formulation of noise elimination problem is provided. The description of the algorithms used to prepare data for neural network training is given. The method of polygon meshes filtering by the means of a bilateral filter, the method of principal components for reducing the dimension of data, the k-means clustering algorithm, the algorithm for updating vertex positions by updated face normals are employed. Details of a fully

connected feedforward neural network implementation are described. The results of the study are outlined. The analysis of the findings is carried out utilizing the quality metrics of the Hausdorff distance and the average value of the angle between the normals of polygon meshes with and without noise.

**Keywords:** neural networks, data science, polygon meshes, mesh denoising, three-dimensional scanning, bilateral filter.

**For citation:** Rotova O.M., Pivovarova N.V. Neural network denoising in polygon meshes. . *Modeling, Optimization and Information Technology*. 2022;10(1). Available from: <https://moitvvt.ru/ru/journal/pdf?id=1093> DOI: 10.26102/2310-6018/2022.36.1.013 (In Russ).

## Введение

С развитием вычислительных мощностей трехмерные технологии стали весьма востребованы в различных областях деятельности. 3D-моделирование применяется в таких сферах, как наука, медицина, промышленность, архитектура, киноиндустрия, создание компьютерных игр, дополненная и виртуальная реальность. Важной задачей современных 3D-технологий является создание цифровых двойников, представляющих собой виртуальные копии объектов реального мира. Математические модели цифрового двойника и объекта реального мира в основных характеристиках идентичны. Для создания цифровых двойников используют технологии трехмерного сканирования объектов. Фотограмметрия представляется одним из примеров таких технологий.

Фотограмметрия в терминах создания цифровых двойников – это процесс создания 3D-моделей из многих различных изображений объекта, сфотографированного с разных ракурсов. 3D-модели представляют в виде полигональных сеток. Полученные с помощью фотограмметрии необработанные модели неизбежно содержат аддитивный шум от различных источников, уровень которого зависит от разрешающей способности используемых камер и точности алгоритмов построения 3D-модели. Процесс устранения шума используется в качестве необходимого шага постобработки 3D-модели после сканирования. Сложной задачей является устранение шума с полигональной сетки с сохранением при этом соответствующих характеристик математической модели объекта реального мира и его цифрового двойника.

### Обзор подходов к решению задачи устранения шума в полигональных сетках Методы, основанные на фильтрации

Билатеральный фильтр, являющийся одним из основных фильтров для шумоподавления в двумерных изображениях, предложен в [1]. Это нелинейный фильтр сглаживания изображения с сохранением четких границ. Данный фильтр широко используется при обработке фотографий.

В [2] было введено определение билатерального фильтра для полигональных сеток, которое основывается на идее билатерального фильтра для изображений. Основная идея заключается в том, чтобы определить локальное пространство соседних компонент для каждой вершины, используя касательную плоскость к полигональной сетке в вершине. Фильтр применяется к одной вершине за один раз, вычисляя смещение и обновляя ее положение. Смещение вершин происходит по направлению нормали к касательной плоскости. Недостатком классической билатеральной фильтрации для полигональных моделей является сложность обработки граничных вершин.

Для расширения возможностей билатерального фильтра с целью обработки изображений в [3, 4] определен совместный билатеральный фильтр. Основная идея этого фильтра состоит в том, что фильтрующие веса могут быть определены с использованием разности интенсивности от другого изображения, называемого руководствующим, а не входным изображением.

В [5] был обобщен подход по использованию совместного билатерального фильтра для полигональных сеток. Идея заключается в том, чтобы сначала классифицировать вершины в соответствии с их пространственными признаками, такими как угол, ребро и плоские области, а затем применять определенные стратегии шумоподавления к каждому типу вершин. Для этого используется направляющий компонент совместного билатерального фильтра. Такой алгоритм справляется с устранением шума на граничных вершинах лучше, чем классический алгоритм билатеральной фильтрации.

В [6, 7] представлены различные интерпретации алгоритмов билатеральной и совместной билатеральной фильтрации, которые обеспечивают решение задачи устранения шума быстрее, но не повышают при этом качество результата.

Рассматривая методы анизотропной фильтрации полигональных сеток для устранения шума на поверхности, можно выделить ряд их преимуществ и недостатков. К преимуществам таких методов можно отнести простоту реализации, интерпретируемость результатов и получение лучшего качества устранения шума по сравнению с более ранними методами изотропной фильтрации. К недостаткам относятся необходимость подбора параметров для каждой отдельной полигональной сетки и уровня шума для получения приемлемого результата, а также неспособность в некоторых случаях справиться с геометрическими особенностями топологии полигональных сеток.

### **Методы, основанные на решении задачи оптимизации.**

В [8] предложен метод устранения шума на поверхности 3D-модели, основанный на решении задачи оптимизации. Данный метод максимизирует плоские области модели и постепенно удаляет шум, сохраняя геометрические особенности. Он основан на построении дискретного дифференциального оператора для произвольных треугольных сеток, который является устойчивым по отношению к вырожденным триангуляциям. Метод дает результаты лучше, чем при применении методов билатеральной фильтрации, даже при наличии высокого уровня шума. Метод достаточно прост в реализации и не требует дополнительных параметров, таких как уровень шума на модели, радиус действия (как в методах фильтрации).

В [9] предложен другой двухступенчатый метод, основанный на сглаживании путем тензорного умножения. В отличие от других традиционных методов, этот метод использует полученный специальным образом тензор нормалей для вычисления гладких поверхностей. Данный метод стабилен при различных типах и уровнях шума, но требует отдельного подбора количества итераций для каждой полигональной модели, а также предварительного задания характеристики шума на поверхности, что само по себе является достаточно сложной задачей.

В [10] представлен нелокальный метод фильтрации нормалей низкого ранга для шумоподавления в сетке. Исследуя геометрическое сходство между локальными поверхностными наборами на 3D-сетках в виде полей нормалей, авторы разработали модель восстановления, которая фильтрует векторы нормалей с помощью поверхностных наборов. С помощью дескриптора ковариации нормалей анализируется сходство между поверхностными наборами. Из схожих поверхностных наборов в матрицу полей нормалей извлекают векторы нормалей для рангового анализа. Далее задача устранения шума формулируется как задача восстановления матрицы низкого ранга, основанная на том, что ранг матрицы высок для необработанных сеток с шумом, но может быть значительно уменьшен для сеток без аддитивного шума, нормальные векторы которых через схожие поверхностные наборы должны быть более сильно

коррелированы. Данный подход стабилен при низком уровне шума, но плохо работает при высоком уровне шума, а также плохо справляется с резкими границами поверхностей полигональных моделей.

К достоинствам рассмотренных методов по устранению шума на 3D-моделях можно отнести получение лучшего результата, чем при применении методов фильтрации. Однако, методы, описанные в [9, 10], все еще требуют в качестве входных данных параметров шума и локализации поверхности.

### **Методы с использованием нейронных сетей.**

Впервые авторами [11] был сформулирован основанный на обучающих данных подход к устранению шума в 3D-моделях. Ключевая идея метода состоит в том, чтобы формализовать процесс шумоподавления с каскадными радиально-базисными регрессионными функциями с помощью нейронных сетей, и изучить эти функции в наборе с зашумленными полигональными сетками и теми же сеткам, но без шума. Каждая регрессионная функция порождает нормаль для заданной выходной грани полигональной сетки и отправляет результат в качестве входных данных следующей регрессионной функции. Данный подход обеспечивает хорошие результаты для разных уровней шума и разных его типов, показывает лучшие результаты по сравнению со всеми рассмотренными выше методами. Основное ограничение заключается в том, что точность реконструкции важных деталей в значительной степени зависит от того, были ли они изначально включены в обучающую выборку.

В исследовании [12] используется двухступенчатый подход к устранению шума на основе сверточных нейронных сетей и преобразования полигональных моделей в воксельный формат, где первым шагом выполняется поиск направляющей нормали с помощью сверточных нейронных сетей и вокселизации полигональной сетки, а вторым шагом применяется совместный билатеральный фильтр на основе найденной направляющей нормали. Процесс извлечения признаков из обучающих данных является очень длительным и ресурсоемким, потому что происходит процесс перевода 3D-модели из одного формата – полигональной сетки, в другой формат – воксельное представление. Также для получения хорошего результата необходимо вручную подбирать параметры для каждой модели, такие как, например, количество итераций и параметры совместного билатерального фильтра.

Для разработки метода устранения шума в полигональных сетках, полученных после применения фотограмметрии, использовалась идея подхода на основе нейронных сетей [11]. Для того, чтобы сгладить полигональную сетку и избавиться от шума, необходимо с помощью билатеральных фильтров с разными параметрами составить дескрипторы для каждой нормали грани 3D-модели. Затем с помощью кластеризации распределить их на разные группы, представляющие различные геометрические особенности. Наконец, следует применить многослойную искусственную нейронную сеть прямого распространения для получения обновленных нормалей из указанных дескрипторов. С помощью обновленных нормалей необходимо обновить позиции координат вершин полигональной сетки.

Исследование эффективности использования нейронных сетей для устранения шума в полигональных сетках является целью настоящей статьи. Для этого ставится задача построения и исследования с применением многослойной нейронной сети модели, позволяющей переходить от полигональной сетки с шумом к полигональной сетке без шума.

## Материалы и методы

Для обучения нейронных сетей и проведения вычислительных экспериментов использовались данные, собранные авторами работы [11]. Набор данных содержит в себе 84 пары обучающих 3D-моделей – с шумом и без шума. В качестве моделей с шумом используются модели с искусственно сгенерированным шумом, а также результаты сканирования с помощью сканеров Microsoft Kinect v1, Microsoft Kinect v2 и Microsoft Kinect Fusion.

### Математическая постановка

Обозначим треугольную полигональную сетку  $M = \{V, F\}$ , где  $V = \{v_i\}_{i=1}^n$  - набор вершин;  $F = \{f_i\}_{i=1}^{n_f}$  - набор граней. Каждая вершина  $v_i$  определена в пространстве с помощью трех координат

$$v_i = [x_i, y_i, z_i]^T, \quad (1)$$

а каждая из треугольных граней  $f_i$  - с помощью образующих ее трех вершин

$$f_j = \{v_{j1}, v_{j2}, v_{j3}\}. \quad (2)$$

Треугольная грань  $f_j$  может быть описана с помощью своего центраида

$$c_j = \frac{(v_{j1} + v_{j2} + v_{j3})}{3} \quad (3)$$

и соответствующей нормали

$$n_{c_j} = \frac{(v_{j2} - v_{j1}) \times (v_{j3} - v_{j1})}{\|(v_{j2} - v_{j1}) \times (v_{j3} - v_{j1})\|}, \quad (4)$$

где  $v_{j1}, v_{j2}, v_{j3}$  – вершины, определяющие грань  $f_j$ .

Пусть полигональная сетка  $\tilde{M}$ , содержащая набор вершин  $\tilde{V}$ , состоящий из  $n$  элементов, и набор треугольных граней  $F$ , состоящий из  $n_f$  элементов, определяет полигональную сетку с шумом. В таком случае координаты вершин  $\tilde{M}$  определяет формула

$$\tilde{V} = V + \varepsilon, \quad (5)$$

где  $\varepsilon$  – смещение координат вершин сетки относительно сетки без шума, которое необходимо минимизировать для решения задачи устранения шума. Для каждой вершины полигональной сетки, из (5) следует выражение

$$\tilde{v}_i = v_i + \varepsilon_i. \quad (6)$$

Координаты вектора нормали грани зависят от координат вершин, которые ее составляют (4). Для устранения шума в полигональных сетках сделан переход от определения шума через координаты вершин (6) к его определению через координаты векторов нормалей граней

$$\tilde{n}_{c_j} = n_{c_j} + \varepsilon_{n_j}. \quad (7)$$

Здесь  $\tilde{n}_{c_j}$  – вектор нормали грани  $f_j$  сетки с шумом;  $\varepsilon_{n_j}$  – вектор шума или вектор смещения координат вектора нормали грани сетки без шума.

Таким образом, в задаче устранения шума необходимо из нормалей  $\tilde{n}_{c_j}$  полигональной сетки с шумом получить нормали  $n_{c_j}$  полигональной сетки без шума.

### Билатеральный фильтр

Итеративный процесс обновления нормалей полигональной сетки с помощью применения билатерального фильтра описывает формула

$$\mathbf{n}_i^{k+1} = \Lambda \left( \sum_{f_j \in F} A_j W_s(\|\mathbf{c}_i - \mathbf{c}_j\|) W_r(\|\mathbf{n}_i^k - \mathbf{n}_j^k\|) \mathbf{n}_j^k \right), \quad (8)$$

где  $F$  – набор граней полигональной сетки;  $\mathbf{n}_i^k$  – нормаль грани  $f_i$  на итерации  $k$ ;  $\mathbf{c}_i$  – центроид грани  $f_i$ ;  $A_i$  – площадь поверхности грани  $f_i$ ;  $W_\sigma(x) = e^{-x^2/(2\sigma^2)}$  – функция Гаусса с заданным стандартным отклонением  $\sigma$ ;  $\Lambda(x)$  – операция нормализации вектора  $x$ .

Согласно [7], величина  $\sigma_s$  для функций Гаусса  $W_s(x)$  выбирается в соответствии со средней длиной ребра полигональной сетки, а величина  $\sigma_r$  для  $W_r(x)$  может быть любым вещественным числом: чем больше эта величина, тем больше значение шума, которое успешно обрабатывает билатеральный фильтр.

### Составление дескрипторов

Пусть имеется набор параметров  $P$ , который содержит пары параметров стандартного отклонения для билатерального фильтра  $\{(\sigma_{s_j}, \sigma_{r_j})\}_{j=1}^L$ , где  $L$  – число таких пар.

Дескриптор  $S_i$  для грани  $f_i$  составляется из нормалей, полученных с помощью билатерального фильтра по формуле (8):

$$S_i = (\mathbf{n}_i(\sigma_{s_1}, \sigma_{r_1}), \mathbf{n}_i(\sigma_{s_2}, \sigma_{r_2}), \dots, \mathbf{n}_i(\sigma_{s_L}, \sigma_{r_L})). \quad (9)$$

Дескриптор  $S_i$  для грани  $f_i$  содержит в себе различные положения нормалей, полученных с помощью применения фильтров с разными заданными параметрами.

Пусть дескриптор  $S_i$  содержит  $d$  отфильтрованных нормалей для грани  $f_i$ :

$$S_i = \{\mathbf{n}_1, \mathbf{n}_2, \dots, \mathbf{n}_d\}. \quad (10)$$

Для того, чтобы сделать дескриптор инвариантным к вращению, необходимо составить матрицу  $T$

$$T = \sum_{j=1}^d \mathbf{n}_j \times \mathbf{n}_j^T. \quad (11)$$

Три собственных вектора матрицы  $T$ , отсортированные по величинам ее собственных значений, определяют матрицу вращений  $R$ . С помощью умножения каждой отфильтрованной нормали  $\mathbf{n}_j$  дескриптора  $S_i$  на матрицу  $R^{-1}$ , дескриптор  $S_i$  «выравнивается», то есть, становится инвариантным к вращению.

### Понижение размерности и кластеризация дескрипторов

Кластеризовать полученные дескрипторы граней необходимо, чтобы определить различные геометрические структуры, которые они представляют – плоские, искривленные, угловые поверхности и другие, и обучить свою нейронную сеть для каждого из кластеров. Для того, чтобы сравнивать два объекта, необходимо иметь критерий для сравнения. Как правило, таким критерием является «расстояние» между объектами. В качестве метрики расстояния было выбрано евклидово расстояние. На Рисунке 1 изображены результаты кластеризации дескрипторов граней для 3D-модели: одинаковыми тонами обозначена принадлежность граней к одинаковым кластерам.

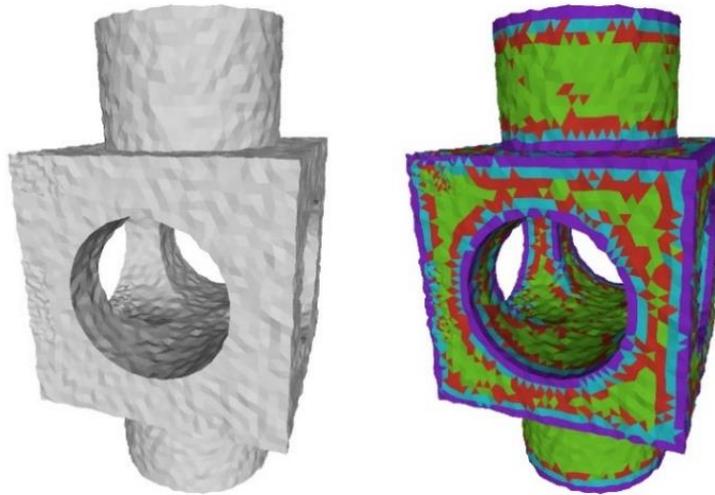


Рисунок 1 – Визуализация кластеризации дескрипторов граней для 3D-модели с шумом  
 Figure 1 – Visualization of clustering of face descriptors for 3D model with noise

В качестве алгоритма кластеризации использовался метод  $k$ -средних. Входными данными для алгоритма являются набор из  $n$  наблюдений  $X = \{x_1, x_2, \dots, x_n\}$  и  $k$  – требуемое число кластеров. Введем обозначение  $S = \{S_1, S_2, \dots, S_n\}$ . Алгоритм  $k$ -средних требует решения задачи

$$\min_S \sum_{i=1}^k \sum_{x \in S_i} \rho(x, \mu_i)^2, \quad (12)$$

где  $\mu_i$  – центры кластеров;  $\rho(x, \mu_i)$  – расстояние между  $x$  и  $\mu_i$ .

Понижение размерности дескрипторов  $S$  необходимо, чтобы использовать метод кластеризации  $k$ -средних. При больших размерностях данных алгоритм дает неудовлетворительные результаты (из-за простоты метода). В качестве алгоритма понижения размерности данных использовался метод главных компонент, в котором учитывается значительное количество информации при минимальных потерях [13].

### Многослойная нейронная сеть прямого распространения

Для решения задачи устранения шума в полигональной сетке для каждого кластера, полученного на предыдущем этапе, используется многослойная искусственная нейронная сеть (ИНС) [14]. В качестве входных данных для обучения ИНС используются дескрипторы граней сеток с шумом и нормали граней исходных сеток без шума из обучающего набора данных. После обучения ИНС для каждого кластера дескрипторов в качестве результата работы нейронная сеть должна, в идеале, выдавать обновленные нормали граней полигональных сеток из обучающего набора данных, дескрипторы которых подавались на вход нейронной сети. Процесс выполняется итерационно, каскадами (Рисунок 2).



Рисунок 2 – Каскадная схема устранения шума  
Figure 2 – Cascaded noise elimination scheme

Полносвязная многослойная нейронная сеть реализована на языке *Python* с использованием фреймворка *PyTorch*. Обучение нейронных сетей происходило с использованием оптимизационного алгоритма *Adam* (adaptive moment estimation) [15] и среднеквадратичной функцией ошибки. В качестве функции активации была выбрана функция *ReLU* (Rectified Linear Unit) [16]. Архитектура используемой нейронной сети в данной работе представлена на Рисунке 3.

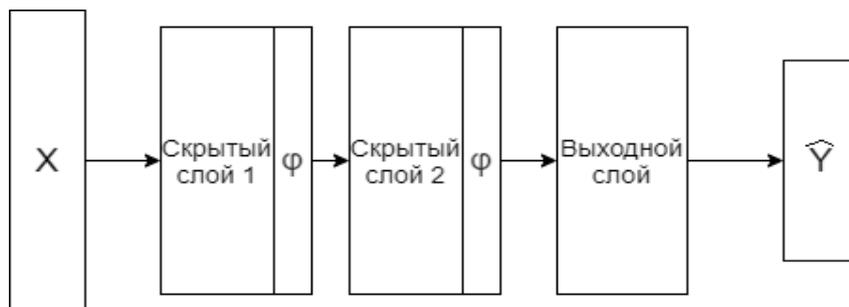


Рисунок 3 – Архитектура используемой нейронной сети  
Figure 3 – Neural network architecture

Размерность входных данных  $X - (N_f \times 3L)$ , где  $N_f$  – число граней полигональной сетки;  $L$  – число пар параметров для билатеральных фильтров для составления дескрипторов. Размерность выходных данных  $\hat{Y} - (N_f \times 3)$ . Число скрытых нейронов для первого скрытого слоя – 100, для второго – 50, для последнего выходного слоя – 3.

В качестве начальных параметров стандартного отклонения для билатерального фильтра выбраны следующие:

- $\sigma_s = \{1l; 2l\}$ , где  $l$  – средняя длина ребра полигональной сетки;
- $\sigma_r = \{0,1; 0,2; 0,35; 0,5; \infty\}$ .

Для обучения каждой нейронной сети каждого из кластеров, обучающий набор данных, входящий в соответствующий кластер, делился на две группы в соотношении 85 % на 15 %: обучающие данные, с помощью которых происходило непосредственно обучение нейронной сети, и валидационные данные, с помощью которых оценивалось качество обучения нейронной сети.

### Алгоритм обновления позиций вершин

Результатом работы многослойной нейронной сети являются новые нормали  $\hat{n}_i$  для грани  $f_i$ , с помощью которых обновляются координаты вершин исходной

зашумленной полигональной сетки. Для этого используется алгоритм, определенный в [7].

Итеративный процесс обновления координат вершин по отфильтрованным нормальям граней определяется формулой

$$v_i^{k+1} = v_i^k + \frac{1}{3|\Omega(v_i)|} \sum_{f_l \in \Omega(v_i)} \sum_{e_{ij} \in f_l} (\hat{n}_l \cdot (v_j^k - v_i^k)) \hat{n}_l. \quad (13)$$

Здесь  $\Omega(v_i)$  – множество соседних вершин для вершины  $v_i$ , которые связаны одним ребром;  $e_{ij}$  – ребро треугольной грани  $f_l$  с вершинами  $v_i, v_j$ ;  $k$  – номер текущей итерации.

### Результаты

Для того, чтобы оценивать качество работы методов устранения шума на полигональных сетках, необходимо ввести метрики качества. Во всех рассмотренных ранее работах [9, 10, 11, 12] в качестве метрик используются расстояние Хаусдорфа и среднее значение угла между нормальями обновленной полигональной сетки и исходной сетки без шума.

Расстояние Хаусдорфа для полигональных сеток было введено в [17]. Данная метрика рассчитывается по формуле

$$d_s(S, S^{\setminus}) = \max[d(S, S^{\setminus}), d(S^{\setminus}, S)], \quad (14)$$

где  $S, S^{\setminus}$  – поверхности сравниваемых полигональных сеток;  $d(S, S^{\setminus})$  – расстояние между поверхностями  $S$  и  $S^{\setminus}$ , которое определяется формулами

$$d(S, S^{\setminus}) = \max_{p \in S} d(p, S^{\setminus}), \quad (15)$$

$$d(p, S^{\setminus}) = \min_{p^{\setminus} \in S^{\setminus}} \|p - p^{\setminus}\|_2. \quad (16)$$

Здесь  $p, p^{\setminus}$  – точки поверхностей  $S, S^{\setminus}$  соответственно.

Значение угла  $\varphi$  в градусах между нормалью обновленной полигональной сетки и нормалью исходной полигональной сетки без шума рассчитывается по формуле

$$\varphi = \arccos \frac{(n_1 \cdot n_2)}{|n_1| |n_2|}, \quad (17)$$

где  $n_1, n_2$  – соответствующие нормали.

Для вычисления значения метрики среднего угла между нормальями двух полигональных сеток, необходимо вычислить все значения углов между соответствующими нормальями исходной сетки без шума и обновленной сетки по формуле (17) и взять среднее арифметическое от этих значений.

Использовались одинаковые исходные полигональные сетки, содержащие три типа синтетического шума, различающегося по величине смещения координат. В Таблице 1 представлены результаты нейросетевого устранения шума.

В Таблице 1 приняты следующие обозначения:  $n_f$  – число граней;  $\varphi_0$  – метрика среднего значения угла между нормальями сетки с шумом и нормальями исходной сетки без шума;  $\varphi_1, \varphi_2, \varphi_3$  – метрики среднего значения угла между нормальями сетки с шумом и нормальями обновленной сетки после каскадов 1, 2 и 3 соответственно;  $d_0$  – значение метрики расстояния Хаусдорфа между полигональной сеткой с шумом и исходной полигональной сеткой без шума;  $d_1, d_2, d_3$  – значения метрик расстояния Хаусдорфа

между полигональной сеткой с шумом и обновленной полигональной сеткой после каскадов 1, 2 и 3 соответственно.

Таблица 1 – Значения метрик качества для полигональных сеток с разным уровнем шума.  
Table 1 – Values of quality metrics for polygonal grids with different noise levels.

| Название объекта | $n_f$  | $\varphi_0, ^\circ$ | $\varphi_1, ^\circ$ | $\varphi_2, ^\circ$ | $\varphi_3, ^\circ$ | $d_0$ | $d_1$ | $d_2$ | $d_3$ |
|------------------|--------|---------------------|---------------------|---------------------|---------------------|-------|-------|-------|-------|
| block_n1         | 17550  | 10,6                | 3,2                 | 2,1                 | 1,9                 | 0,29  | 0,2   | 0,17  | 0,16  |
| block_n2         | 17550  | 19,9                | 5,3                 | 2,9                 | 2,6                 | 0,51  | 0,32  | 0,26  | 0,25  |
| block_n3         | 17550  | 27,8                | 8,2                 | 5,1                 | 4,4                 | 0,82  | 0,49  | 0,46  | 0,45  |
| nicolo_n1        | 99994  | 12,3                | 3,7                 | 3,7                 | 3,7                 | 0,01  | 0,01  | 0,01  | 0,01  |
| nicolo_n2        | 99994  | 22,7                | 4,7                 | 4,3                 | 4,3                 | 0,03  | 0,02  | 0,01  | 0,01  |
| nicolo_n3        | 99994  | 30,9                | 6,1                 | 5,3                 | 5,1                 | 0,04  | 0,02  | 0,02  | 0,02  |
| Chinese_lion_n1  | 100000 | 12,5                | 5,9                 | 5,8                 | 5,9                 | 0,39  | 0,38  | 0,37  | 0,38  |
| Chinese_lion_n2  | 100000 | 23,1                | 7,2                 | 6,9                 | 6,9                 | 0,84  | 0,53  | 0,47  | 0,45  |
| Chinese_lion_n3  | 100000 | 31,5                | 8,8                 | 8,3                 | 8,2                 | 1,21  | 0,68  | 0,68  | 0,69  |

Результат нейросетевого устранения шума для рассмотренных объектов из Таблицы 1 представлен на Рисунках 4, 5.

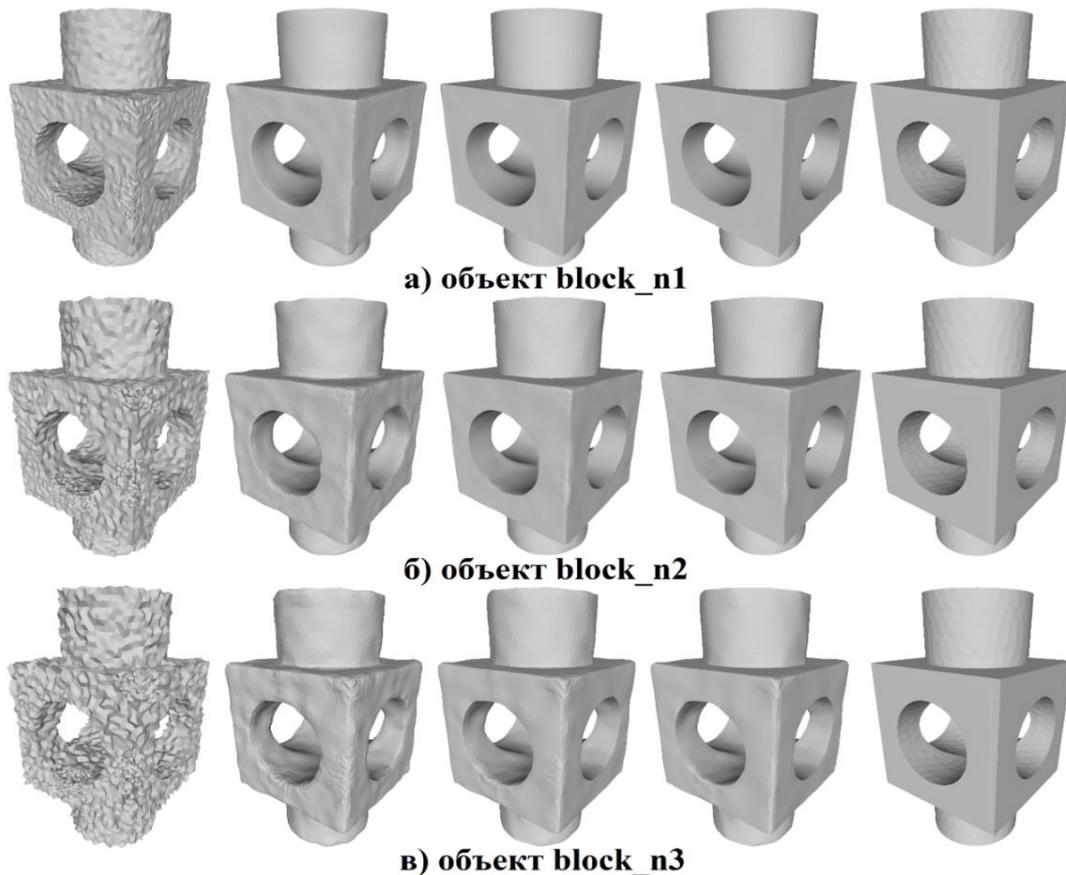


Рисунок 4 – Результат нейросетевого устранения шума в полигональных сетках для объектов класса block. Слева направо: объект с шумом; результаты после каскадов 1, 2, 3; исходный объект без шума

Figure 4 – The result of neural network noise elimination in polygonal grids for objects of the block class. From left to right: object with noise; results after cascades 1, 2, 3; source object without noise

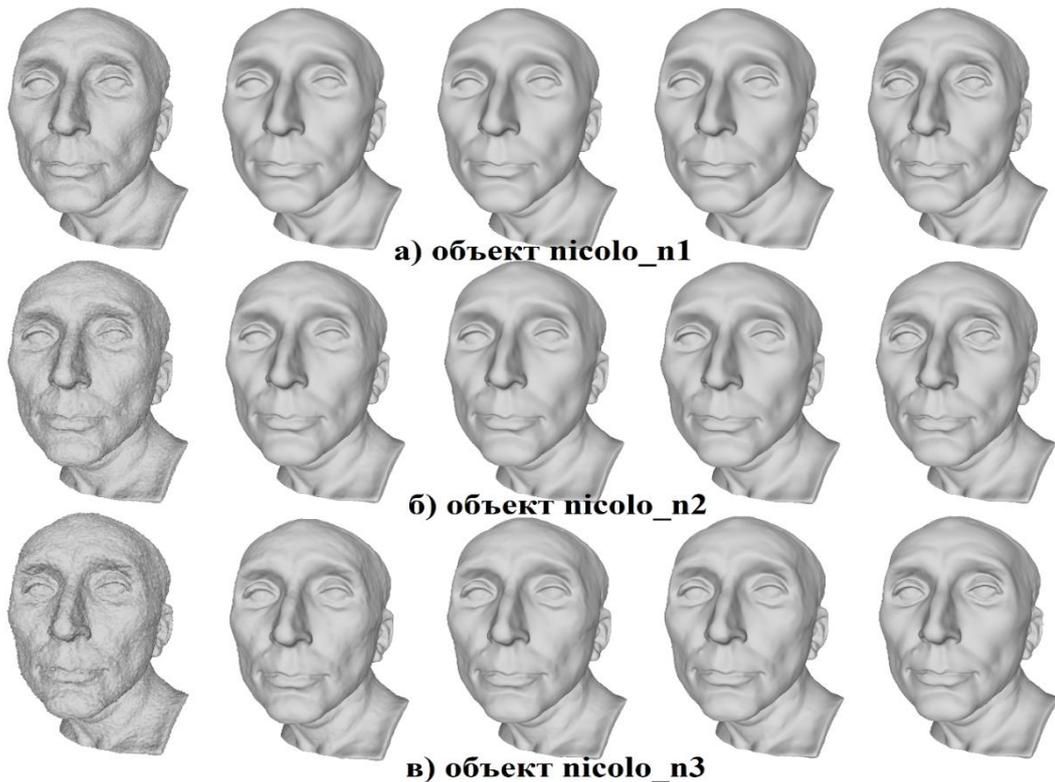


Рисунок 5 – Результат нейросетевого устранения шума в полигональных сетках для объекта nicolo. Слева направо: объект с шумом; результаты после каскадов 1, 2, 3; исходный объект без шума

Figure 5 – The result of neural network noise elimination in polygonal grids for objects of the nicolo class. From left to right: object with noise; results after cascades 1, 2, 3; source object without noise

### Заключение

Использование алгоритмов фильтрации и оптимизации для задачи устранения шума в полигональных сетках, полученных после трехмерного сканирования, в большинстве случаев представляется достаточно сложным и трудоемким процессом, так как необходимо вручную подбирать параметры сглаживания. Для избавления от таких трудностей в данной работе рассмотрено использование технологии нейронных сетей.

Использование полносвязных многослойных нейронных сетей упрощает процесс шумоподавления за счет того, что обученная нейронная сеть самостоятельно отбирает признаки, полученные с помощью билатеральных фильтров с разными параметрами, что позволяет не задавать их вручную. Также предложенный метод не требует знания характеристик шума, который появился в результате сканирования. Однако для обучения нейронных сетей требуются различные 3D-модели с разным типом шума.

Полученные результаты позволяют сделать вывод об эффективности рассматриваемого подхода.

В настоящее время использование нейронных сетей в трехмерных технологиях при обработке облаков точек и полигональных сеток стремительно развивается. На следующем этапе работы над задачей устранения шума в полигональных сетках планируется создание графовой сверточной нейронной сети, которая сможет обрабатывать как локальную структуру полигональной сетки (в пределах окрестности граней), так и глобальную структуру всей 3D-модели, что позволит учитывать локальные и глобальные признаки полигональной сетки одновременно.

## СПИСОК ИСТОЧНИКОВ

1. Tomasi C., Manduchi R. Bilateral filtering for gray and color images. *Computer Vision. Sixth International Conference on IEEE*. 1998;1:839–846.
2. Fleishman S., Drori I., Cohen-or D. Bilateral mesh denoising. *ACM Transactions on Graphics*. 2003;22(3):950–953.
3. Eisemann E., Durand F. Flash photography enhancement via intrinsic relighting. *ACM Transactions on Graphics*. 2004;23(3):673.
4. Petschnigg G., Szeliski R., Agrawala M., Cohen M., Hoppe H., Toyama K. Digital photography with flash and no-flash image pairs. *ACM Transactions on Graphics*. 2004;23(3):664.
5. Zhang W., Deng B., Zhang J., Bouaziz S., Liu L. Guided mesh normal filtering. *Computer Graphics Forum (Special Issue of Pacific Graphics 2015)*. 2015;34:23–34.
6. Sun X., Rosin P.L., Martin R.R., Langbein F.C. Fast and effective feature-preserving mesh denoising. *IEEE Transactions on Visualization and Computer Graphics*. 2007;13(5):925–938.
7. Wei M., Yu J., Pang W.M., Wang J., Qin J., Liu L., Heng P.A. Bi-normal filtering for mesh denoising. *IEEE Transactions on Visualization and Computer Graphics*. 2015;21(1):43–55.
8. He L., Schaefer S. Mesh denoising via l0 minimization. *ACM Transactions on Graphics*. 2013;32(4):64.
9. Yadav S.K., Reitebuch U., Polthier K. Mesh Denoising Based on Normal Voting Tensor and Binary Optimization. *IEEE Transactions on Visualization and Computer Graphics*. 2018;24(8):2366–2379.
10. Li X., Zhu L., Fu C.-W., Heng P.-A. Non-Local Low-Rank Normal Filtering for Mesh Denoising. *Computer Graphics Forum*. 2018;37(7):155–166.
11. Wang P.-S., Liu Y., Tong X. Mesh denoising via cascaded normal regression. *ACM Transactions on Graphics*. 2016;35(6):1–12.
12. Wang C., Cheng M., Sohel F., Bennamoun M., Li J. NormalNet: A voxel-based CNN for 3D object classification and retrieval. *Neurocomputing*. 2019;323:139–147.
13. Гантмахер Ф. П. *Теория матриц*. М.: Наука; 1966. 576 с.
14. *Большая российская энциклопедия*. М.: Большая российская энциклопедия, 2004–2017.
15. Kingma D.P., Ba J. Adam: a method for stochastic optimization. *ICLR*. 2015;1:1–15.
16. Agarap A.F. Deep learning using rectified linear units (RELU). *Arxiv*. 2018;1:1–7.
17. Aspert N., Santa-Cruz D., and Ebrahimi T. MESH: Measuring Errors between Surfaces using the Hausdorff Distance. *Proc. IEEE ICME*. 2002;1:705–708.

## REFERENCES

1. Tomasi C., Manduchi R. Bilateral filtering for gray and color images. *Computer Vision. Sixth International Conference on IEEE*. 1998;1:839–846.
2. Fleishman S., Drori I., Cohen-or D. Bilateral mesh denoising. *ACM Transactions on Graphics*. 2003;22(3):950–953.
3. Eisemann E., Durand F. Flash photography enhancement via intrinsic relighting. *ACM Transactions on Graphics*. 2004;23(3):673.
4. Petschnigg G., Szeliski R., Agrawala M., Cohen M., Hoppe H., Toyama K. Digital photography with flash and no-flash image pairs. *ACM Transactions on Graphics*. 2004;23(3):664.

5. Zhang W., Deng B., Zhang J., Bouaziz S., Liu L. Guided mesh normal filtering. *Computer Graphics Forum (Special Issue of Pacific Graphics 2015)*. 2015;34:23–34.
6. Sun X., Rosin P.L., Martin R.R., Langbein F.C. Fast and effective feature-preserving mesh denoising. *IEEE Transactions on Visualization and Computer Graphics*. 2007;13(5):925–938.
7. Wei M., Yu J., Pang W.M., Wang J., Qin J., Liu L., Heng P.A. Bi-normal filtering for mesh denoising. *IEEE Transactions on Visualization and Computer Graphics*. 2015;21(1):43–55.
8. He L., Schaefer S. Mesh denoising via l0 minimization. *ACM Transactions on Graphics*. 2013;32(4):64.
9. Yadav S.K., Reitebuch U., Polthier K. Mesh Denoising Based on Normal Voting Tensor and Binary Optimization. *IEEE Transactions on Visualization and Computer Graphics*. 2018;24(8):2366–2379.
10. Li X., Zhu L., Fu C.-W., Heng P.-A. Non-Local Low-Rank Normal Filtering for Mesh Denoising. *Computer Graphics Forum*. 2018;37(7):155–166.
11. Wang P.-S., Liu Y., Tong X. Mesh denoising via cascaded normal regression. *ACM Transactions on Graphics*. 2016;35(6):1–12.
12. Wang C., Cheng M., Sohel F., Bennamoun M., Li J. NormalNet: A voxel-based CNN for 3D object classification and retrieval. *Neurocomputing*. 2019;323:139–147.
13. Gantmaher F.R. *Matrix theory*. Moscow, Nauka; 1966. 576 p. (In Russ.)
14. *The Great Russian Encyclopedia*. Moscow, The Great Russian Encyclopedia; 2004–2017. (In Russ.)
15. Kingma D.P., Ba J. Adam: a method for stochastic optimization. *ICLR*. 2015;1:1–15.
16. Agarap A.F. Deep learning using rectified linear units (RELU). *Arxiv*. 2018;1:1–7.
17. Aspert N., Santa-Cruz D., and Ebrahimi T. MESH: Measuring Errors between Surfaces using the Hausdorff Distance. *Proc. IEEE ICME*. 2002;1:705–708.

## ИНФОРМАЦИЯ ОБ АВТОРАХ / INFORMATION ABOUT THE AUTHORS

**Ротова Ольга Максимовна**, ООО «С7 Техлаб», аналитик данных, магистрант Московского государственного технического университета имени Н.Э. Баумана, Москва, Российская Федерация.  
*e-mail:* [ol-rtv@yandex.ru](mailto:ol-rtv@yandex.ru)

**Rotova Olga Maksimovna**, ООО «S7 Techlab», data scientist, master student of Bauman Moscow State Technical University, Moscow, Russian Federation.

**Пивоварова Наталья Владимировна**, кандидат технических наук, профессор Московского государственного технического университета имени Н.Э. Баумана, Москва, Российская Федерация.  
*e-mail:* [pivovarova.natasha2013@yandex.ru](mailto:pivovarova.natasha2013@yandex.ru)

**Pivovarova Natalya Vladimirovna**, Candidate of Technical Science, Professor of Bauman Moscow State Technical University, Moscow, Russian Federation.

*Статья поступила в редакцию 12.12.2021; одобрена после рецензирования 22.01.2022; принята к публикации 18.02.2022.*

*The article was submitted 12.12.2021; approved after reviewing 22.01.2022; accepted for publication 18.02.2022.*