

УДК 004.032.26

**Генерирование карт высот с помощью нейронных сетей**

Н. В. Кучур

Ярославский государственный университет им. П. Г. Демидова

E-mail: nikitakuchur@gmail.com

**Аннотация**

Рассматривается задача генерирования реалистичных карт высот. Используются две генеративные модели: вариационный автоэнкодер и генеративно-состязательная сеть.

*Ключевые слова:* нейронные сети, карты высот, генерация ландшафта, генерация изображений, вариационный автоэнкодер, генеративно-состязательная сеть.

**Постановка задачи**

Карта высот (height map) — растровое черно-белое изображение, каждый пиксель которого интерпретируется как высота. Обычно, чем светлее значения пикселя, тем выше эта область и наоборот (рис. 1). Подобные изображения активно используются в компьютерной графике.

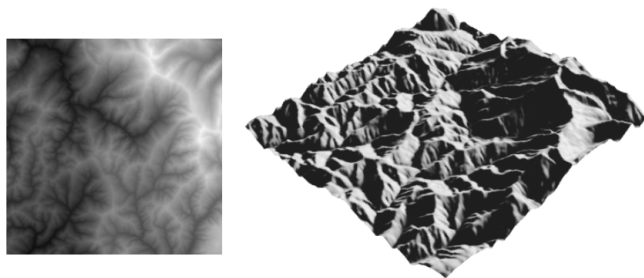


Рис. 1: Карта высот и ее трехмерное представление

Существует множество алгоритмов процедурной генерации карт высот, например, на основе шума Перлина или diamond-square. Однако ни один из них не способен генерировать правдоподобные

© Кучур Н. В., 2020

изображения. Поэтому в качестве альтернативного решения данной задачи можно рассмотреть генеративные нейронные сети.

Задачу можно сформулировать так: необходимо разработать и обучить искусственную нейронную сеть для генерации реалистичных карт высот. В процессе выполнения задачи потребуется:

1. Собрать и подготовить необходимую обучающую выборку.
2. Выбрать топологию сети.
3. Экспериментально подобрать характеристики сети и параметры обучения.
4. Обучить сеть.
5. Проверить результат.

## Результаты предшественников

Несмотря на важность, задача остается малоизученной. Все, что удастся найти по данной тематике, содержится в небольшой статье [1]. В ней были продемонстрированы довольно интересные результаты, но, к сожалению, потенциал нейронных сетей до конца раскрыт не был. Одна из основных проблем — некачественная обучающая выборка. Авторы использовали изображения довольно большого масштаба, которые нарезались с карты высот всей поверхности Земли. Это упущение сильно повлияло на результаты в худшую сторону.

В данной статье мы попытаемся учесть недостатки работы предшественников и лучше подготовить обучающую выборку. Помимо этого, попробуем применить к данной задаче ещё одну интересную генеративную модель — вариационный автоэнкодер.

## Подготовка к работе

Для реализации и обучения нейронной сети был выбран язык программирования Python и библиотека машинного обучения PyTorch (см. [2]). Обучающая выборка была собрана с помощью ресурса Tangram Heightmapper (см. [3]). В итоге вышло около 60 тыс. изображений рельефа, размером 64 на 64 пикселя (рис. 2).



Рис. 2: Пример изображений из обучающей выборки

## Вариационный автоэнкодер

Вариационный автоэнкодер (variational autoencoder, VAE) — модифицированный автоэнкодер, обладающий необходимыми свойствами, которые обеспечивают генеративный процесс. Чтобы подробнее познакомиться с тем, как работает VAE, см. [4] и [5].

Для генеративной модели необходимо, чтобы выполнялись следующие свойства:

- непрерывность: две близкие точки в латентном пространстве не должны давать два совершенно разных результата после их декодирования;
- полнота: точка, выбранная из латентного пространства, при декодировании должна давать значимый результат. То есть мы ожидаем от сети, что генерируемые значения будут похожи на нашу обучающую выборку.

Реализованная сеть для задачи генерации реалистичных карт высот изображена на рис. 3.

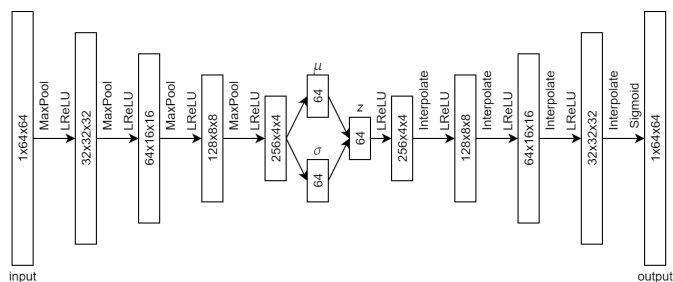


Рис. 3: Архитектура VAE

Ошибка в данном типе сетей рассчитывается с помощью расстояния Кульбака-Лейблера (Kullback–Leibler divergence, KLD):

$$L = d(x, x') + KL[N(\mu, \sigma) \parallel N(0, 1)],$$

где  $d$  — функция для расчета ошибки между входными и выходными значениями,  $KL$  — расстояние Кульбака-Лейблера между имеющимся распределением в латентном пространстве и стандартным нормальным.

Для обучения используется оптимизационный алгоритм Adam (adaptive moment estimation) (см. [6]). В качестве функции расстоя-

ния между входными и выходными значениями была выбрана средняя квадратическая ошибка (mean square error, MSE). Скорость обучения 0.001, размер одного батча 100 и количество эпох 40. На рис. 4 можно увидеть, как в процессе обучения изменялась ошибка.

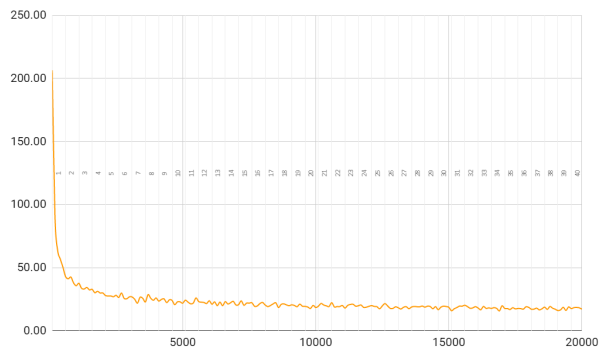


Рис. 4: График изменения ошибки в процессе обучения

После обучения были получены довольно неплохие результаты. На рис. 5 можно увидеть примеры восстановления изображений, а на рис. 6 — примеры генерации.

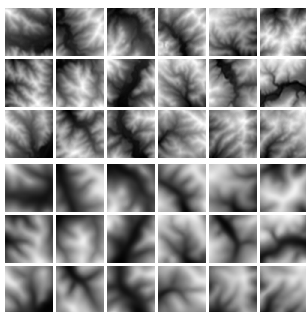


Рис. 5: Восстановление изображений (сверху оригинальные, снизу восстановленные)

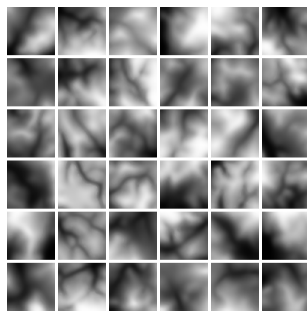


Рис. 6: Пример генерации изображений

Благодаря тому, что сеть обладает свойством непрерывности, можно производить интерполяцию между разными картами высот (рис. 7).

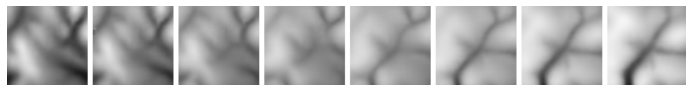


Рис. 7: Интерполяция

На рис. 8 можно увидеть трехмерное представление карт высот, которые были сгенерированы с помощью вариационного автоэнкодера.

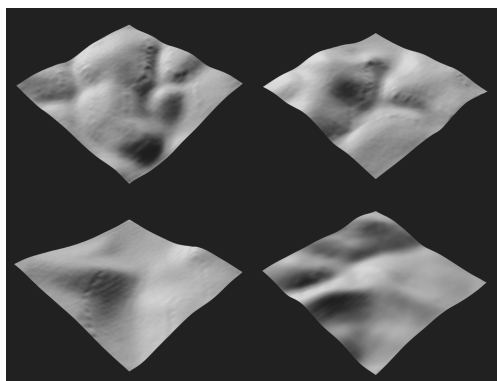


Рис. 8: Трехмерное представление карт высот

К сожалению, как и большинство VAE сетей, эта обладает одним серьезным недостатком: итоговые изображения получаются довольно размытыми. В связи с этим было решено попробовать другую архитектуру нейронной сети.

### Генеративно-сопоставительная сеть

Генеративно-сопоставительная сеть (generative adversarial network, GAN) — архитектура искусственной нейронной сети, состоящей из двух подсетей: генератора и дискриминатора, настроенных на работу друг против друга. Для ознакомления с данным типом сетей см. [7].

Архитектура генератора и дискриминатора представлены на рис. 9 и рис. 10 соответственно.

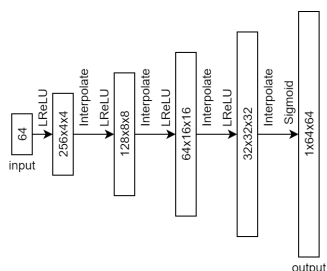


Рис. 9: Архитектура генератора

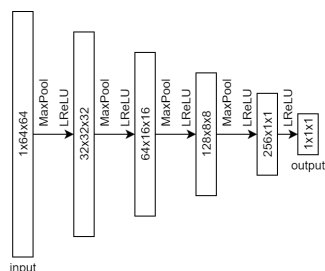


Рис. 10: Архитектура дискриминатора

Для обучения также был использован оптимизационный алгоритм Adam (подробнее см. [6]). В качестве функции расстояния между входными и выходными значениями была выбрана бинарная кросс-энтропия (binary cross-entropy, BSE). Скорость обучения 0.001 для обеих сетей, размер одного батча 100 и количество эпох 100. На рис. 11 можно увидеть, как в процессе обучения изменялись ошибки дискриминатора и генератора.

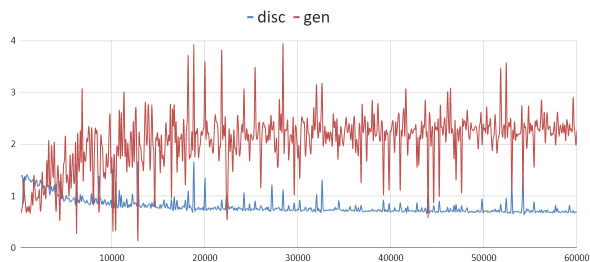


Рис. 11: График изменения ошибок в процессе обучения

После обучения данной сети были получены очень хорошие результаты. Сгенерированные изображения (рис. 12) выглядят очень реалистично.

В этой сети так же, как и в вариационном автоэнкодере, можно производить интерполяцию (рис. 13).

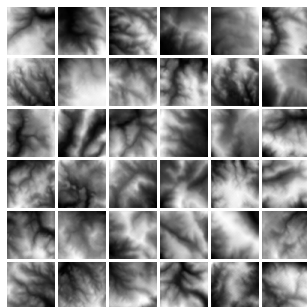


Рис. 12: Генерация изображений

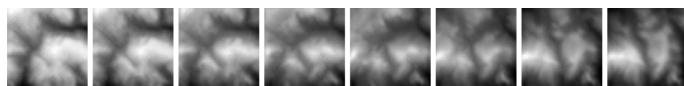


Рис. 13: Интерполяция

На рис. 14 можно увидеть трехмерное представление карт высот, которые были сгенерированы с помощью генеративно-состязательной сети.

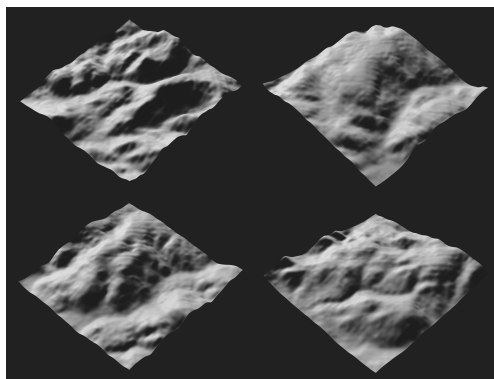


Рис. 14: Трехмерное представление карт высот

**Список литературы**

1. *Beckham C., Pal C.* A step towards procedural terrain generation with GANs. URL: <https://arxiv.org/abs/1707.03383> (visited on 01/15/2020).
2. PyTorch. URL: <https://pytorch.org/> (visited on 01/15/2020).
3. Tangram Heightmapper. URL: <https://tangrams.github.io/heightmapper> (visited on 01/20/2020).
4. *Rocca J.* Understanding Variational Autoencoders (VAEs). URL: <https://towardsdatascience.com/understanding-variational-autoencodersvae-f70510919f73> (visited on 01/20/2020).
5. *Kingma D. P., Welling M.* Auto-Encoding Variational Bayes. URL: <https://arxiv.org/abs/1312.6114> (visited on 01/20/2020).
6. *Kingma D. P., Ba. J.* Adam: A Method for Stochastic Optimization. URL: <https://arxiv.org/abs/1412.6980> (visited on 01/20/2020).
7. *Goodfellow I. J.* [et al.]. Generative Adversarial Networks. URL: <https://arxiv.org/abs/1406.2661> (visited on 01/20/2020).