

СЕГМЕНТАЦИЯ ФАСАДОВ ЗДАНИЙ

Сегментация фасадов с применением
нейронных сетей

Обзор готовых решений с применением ИИ

- ▣ 1) [The bedroom script](#) - проект John Porral, программе задают контур жилья, она создаёт варианты планировок.
- ▣ Планировки объединяются в квартиры и жилые дома.

- ▣ 2) Проект [Spacemaker](#) от компании Autodesk - облачный софт, для генерации квартир. Компания Autodesk приостановила свою деятельность в России (*воспользоваться нельзя*).

- ▣ 3) Архитектор [Abhinav Mishra](#) натренировал нейронные сети генерировать планы и фасады соборов.

- ▣ 4) [The Meters](#) - сервис по поиску недвижимости для покупки и аренды

Обзор готовых решений с применением ИИ

5)

SA lab x daily

Искусственный интеллект и брутализм

Matias del Campo исследует возможность использования генеративно-сопоставительных нейронных сетей в архитектуре. Здесь взят датасет из зданий в стиле брутализм.

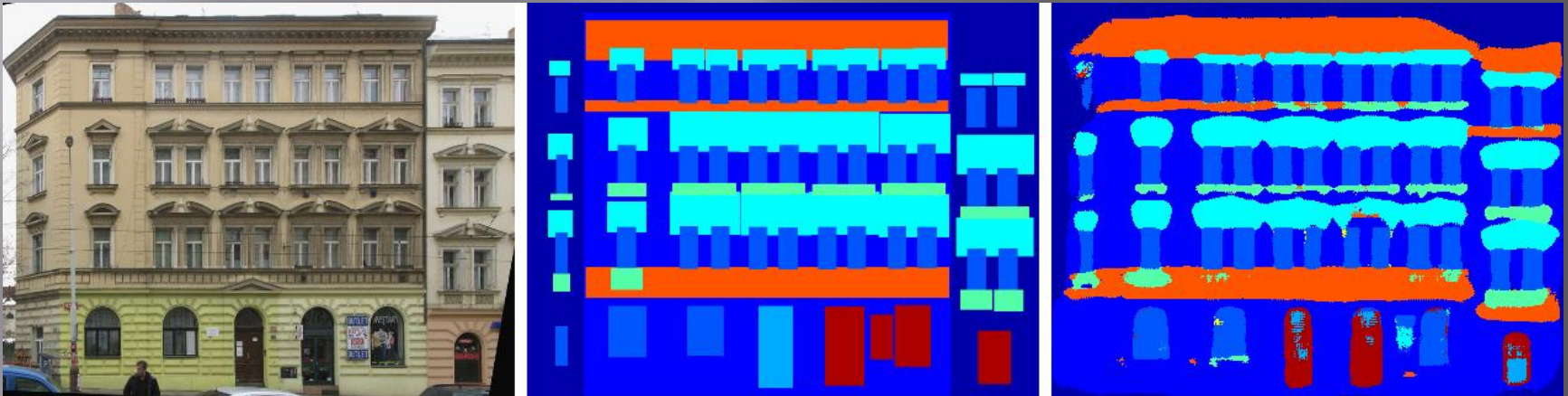
#SAlab_tech



t.me/SAlab_daily/1256 1.6K 👁 Oct 7, 2021 at 18:30

Обзор готовых решений с применением ИИ

- 6) Unet-Segmentation-with-Tensorflow - решение по сегментации выложенное в GitHub
«<https://github.com/prttyflvck0/Unet-Segmentation-with-Tensorflow>»



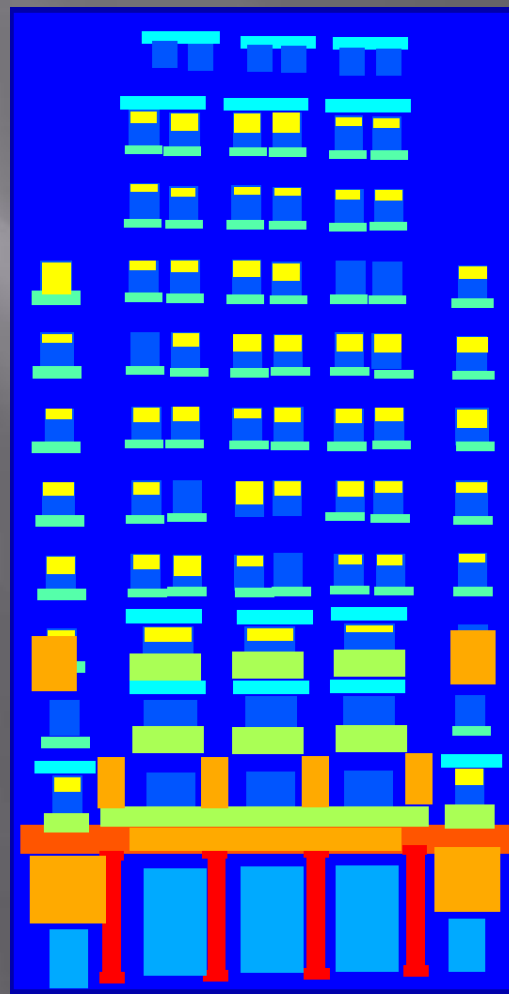
Решение по сегментации фасада здания

Для решения задачи по сегментации фасада здания (выделения сегментов окон) и подсчёта количества окон:

1. Собрать датасет фасадов зданий с размеченными сегментами окон.
2. Построить модель нейронной сети (*Unet*).
3. Натренировать модель сети.
4. Проверить работоспособность модели на тестовых данных.

Датасет

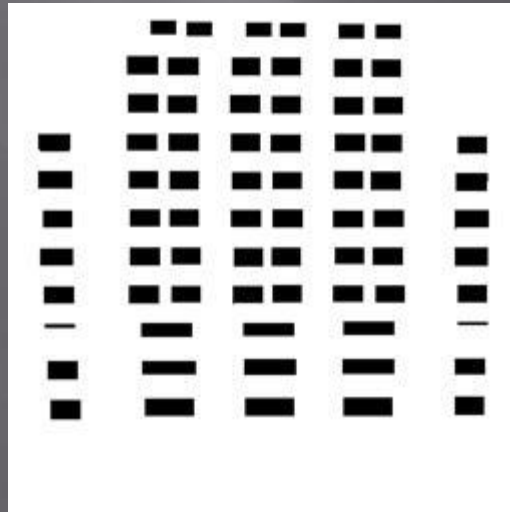
Датасет был взят с Kaggle



Датасет

Данный датасет имеет разметку окон, дверей и прочих элементов фасада.

Из данных изображений масок необходимо убрать лишние элементы, и привести изображение к бинарному виду, также необходимо изменить размер изображений (*чтобы все изображения были одного размера*).



Модель нейронной сети

Для решения задачи сегментации была выбрана архитектура нейронной сети *UNet*.

Для реализации модели сети применялась библиотека *Keras* (*tensorflow*). При обучении *accuracy* достигла всего лишь 0.64

Epoch 10/10

А точность на валидационных данных *val_accuracy: 0.0068*.

Вероятно, в модели сети ошибка, или данные подготовлены плохо.

Модель нейронной сети *resnet50*

Применяя предобученную нейронную сеть *resnet50* с использованием библиотеки *Pytorch*, получил результат лучше.

```
Epoch: 41  train: 100% | ██████████ | 258/258 [01:01<00:00, 4.17it/s, dice_loss - 0.2703, iou_score - 0.5763]  
valid: 100% | ██████████ | 103/103 [00:08<00:00, 11.56it/s, dice_loss - 0.2842, iou_score - 0.5594]
```


Для подсчёта количества окон разработана функция
calc_windows(image)

Функция ищет крайние четыре точки окон (*прямоугольник*) данные (*точки*) заносятся в список. Количество элементов списка и будет количество окон.

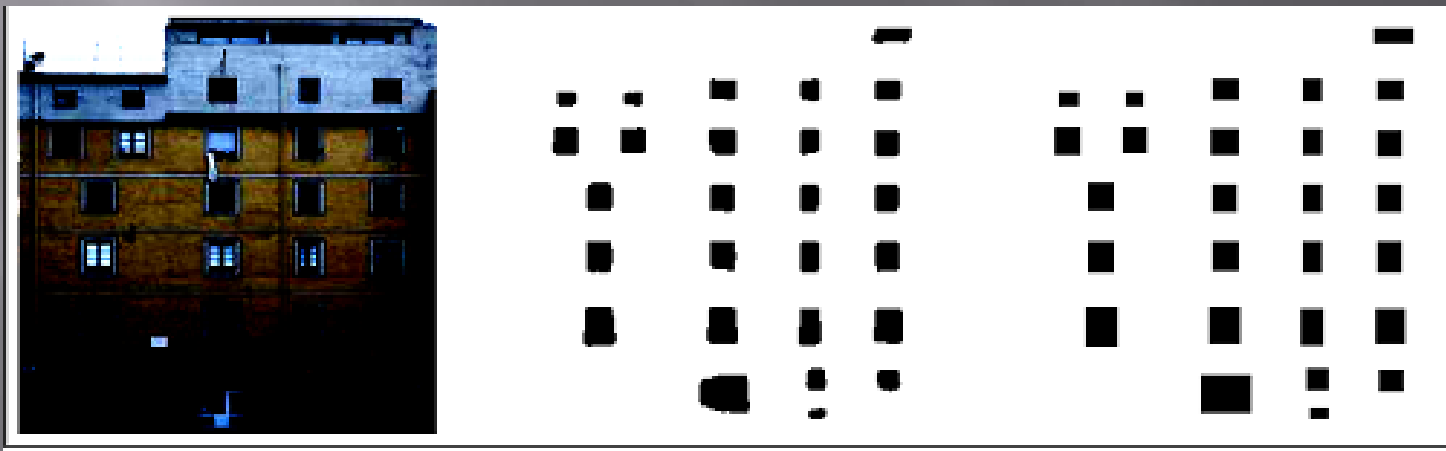
В дальнейшем эту функцию возможно использовать в функции *func_mask()* для подготовки масок для датасета окон (*ниже приведён пример*).

Проверка модели

Точность модели на тестовых данных:

valid: 100% |  | 52/52 [00:40<00:00, 1.29it/s, dice_loss - 0.2808, iou_score - 0.5693]

Точность на проверочной выборке уже 0.5594, что близко к тренировочной выборке. После обработки изображения с помощью функции *calc_windows* и *func_mask* получаем изображение которое можно применять в датасете в качестве масок.



Результат

В ходе работы был собран датасет фасадов окон.

Обучена и применена модель нейронной сети архитектуры *Unet*, с использованием библиотеки *Pytorch*.
Применялась предобученная сеть *resnet50*.

Для подсчёта количества окон разработана функция.
Так же разработана функция формирующая прямоугольники на месте чёрных пятен на выходном изображении.

PS

Опыта работы в данной теме нет, так как недавно окончил ВУЗ после службы в армии. И возрастного сотрудника без опыта никто брать не хочет.

Есть желание работать и развиваться в данной сфере.