Московский государственный университет имени М.В.Ломоносова

Механико-математический факультет

Кафедра Математической теории интеллектуальных систем

Киназаров Темирбек

«Разработка нейросетевых алгоритмов определения пространственных характеристик движущихся объектов»

Курсовая работа

Научный руководитель:

доцент кафедры Математической теории интеллектуальных систем,

кандидат физико-математических наук

Часовских Анатолий Александрович.

Москва, 2020

Оглавление

§ 1 Введение

Обнаружение трехмерных объектов является одной из важнейших задач для систем восприятия автономных транспортных средств. На текущий момент существует несколько различных систем, позволяющих определять расположение трехмерных объектов на плоских изображениях, представленных набором пикселей [1] [2] [3].Во многих из них результат работы представлен в виде двумерной рамки, определяющей положение объекта на кадре. Подобное предствление решения может влечь за собой потерю точности при использовании упомянутых систем в задаче отслеживания перемещения объекта, при заданном наборе кадров видеоряда. Для увеличения точности отслеживания траектории движения существуют модели построения проекции трехмерной рамки вокруг объекта [4] [5]. В рамках настоящей курсовой работы рассматривается задача определения местоположения проекции центра тяжести автомобилей, присутствующих на заданном изображении. Решение указанной задачи позволяет упростить задачу отслеживания перемещения транспортного средства по данным, полученным из видеоряда с некоторой камеры.

В представленной курсовой работе предложено решение подзадачи для указанной задачи, состоящей в определении трёхмерной рамки [6](будем называть ее параллелепипедом), охватывающей транспортные средства на изображении, с последующим определением марки автомобиля. При верном определении трехмерной описывающей рамки для автомобиля на заданном кадре, совместно с маркой автомобиля, можно оценить расположение проекции центра автомобиля с некоторой погрешностью при заданных ограничениях.

Данная задача является актуальной для обнаружения нарушения ПДД автотранспортом путем визуальной идентификации электроннновычислительной техникой.

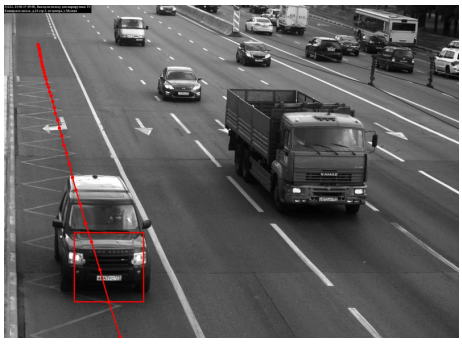

Рис. 1: Пример построения траектории автотранспорта

§ 2 Постановка задачи

При съемке камерой происходит проективное преобразование трехмерной сцены на двумерное изображение. Проекция трехмерной точки на изображение получается по

формуле преобразования координаты исходной точки сцены в координаты пикселя на изображении:

$$z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \left(A[R\,T] \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \right), \tag{2.1}$$

где z_c произвольный масштабный коэффициент. Матрица внутренней калибровки A содержит 5 значимых параметров:

$$A = \begin{pmatrix} \alpha_x & \gamma & u_0 \\ 0 & \alpha_y & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

Эти параметры соответствуют фокусному расстоянию, углу наклона пикселей и принципиальной точке (точка пересечения плоскости изображения с оптической осью, совпадающая с центром фотографии, которая в реальных камерах, как правило, бывает немного смещена из-за оптических искажений). В частности, α_x и α_y соответствуют фокусному расстоянию, измеренному в ширине и высоте пикселя, u_0 и v_0 — координатам принципиальной точки, а $\gamma = \alpha_y * \tan\varphi$, где φ — угол наклона пикселя. R, T (где R — вектор 1 x 3 или матрица 3 x 3 поворота, T — вектор 3 x 1 переноса) — параметры внешней калибровки, определяющие преобразование координат, переводящее координаты точек сцены из трехмерной системы координат в систему координат, связанную с камерой.

В настоящей курсовой работе предполагается, что задана матрица A параметров внутренней калибровки камеры, высота, на которой находится камера и угол наклона. Подобные ограничения наложены на камеры, при помощи которых ведется съемка автотранспорта на дорогах, для автоматического обнаружения нарушений ПДД. Необходимо построить точку описывающую центр нижней грани автотранспорта путем нахождения координат трехмерной рамки, которая охватывает объект. В задачу также входит нахождение направления движения объекта по одному изображению и вывод предполагаемой марки и модели автотранспорта.

§ 3 Описание модели

Модель состоит из трех подзадач:

1. Нахождение трехмерной описывающей рамки для автомобилей в кадре
2. Определение марки автомобилей

3. Определение проекции центра нижней грани описывающего трехмерной рамки в кадре при помощи данных о расположении камеры, а так же используя матрицу камеры.

§ 3.1 Нахождение трехмерной описывающей рамки для автомобилей в кадре

В статье [6] описывается модель, решающая первую подзадачу. Используя информацию о расположении двумерной охватывающей рамки модель находит координаты углов трёхмерной охватывающей рамки в кадре. Авторы статьи выбрали в качестве параметров для описания трехмерной рамки такие параметры, как центр предполагаемой рамки (параллелепипеда) в виде трех координат $T = [t_x, t_y, t_z]$, длин его трех сторон $D = [d_x, d_y, d_z]$ и ориентация $R(\theta, \phi, \alpha)$ выбранных углов Эйлера. Всего выходит 9 степеней свободы.

Первым действтием является пропуск нашего изображения через сеть с обученными весами для нахождения двумерной охватывающей рамки автомобиля в видеоряде [12]. Обозначим (x_{min}, y_{min}) - координаты левой верхней вершины и (x_{max}, y_{max}) - координаты правой нижней вершины двумерной рамки. Если мы знаем размеры нашего объекта, то, установив начало координат в центре трехмерной рамки (после проекции соответствующем центру двумерной рамки), можем вычислить координаты всех восьми вершин параллелепипеда, то есть $[\pm\frac{d_x}{2}, \pm\frac{d_y}{2}, \pm\frac{d_z}{2}]$. Далее предполагается, что проекция нашей трехмерной рамки на изображении должна лежать внутри нашего найденного параллелограмма. Откуда возникают 4 уравнения:

$$x_{min} = \left(K[R\,T] \begin{bmatrix} \frac{d_x}{2} \\ -\frac{d_y}{2} \\ \frac{d_z}{2} \end{bmatrix} \right)_x , \tag{3.1}$$

где $(.)_x$ означает проекцию на вектор x, то есть выделяется только первая координата полученного вектора. В частности x_{min} есть проекция одной из 8-ми точек на изображение.

Этого недостаточно для ограничения девяти степеней свободы (три для перемещения, три для вращения и три для размеров коробки). Существует несколько различных геометрических свойств, которые можно оценить по внешнему виду описывающего прямоугольника, чтобы дополнительно ограничить 3D-рамку.

Первый набор параметров, которые сильно влияют на трехмерную рамку, это ориентация вокруг каждой оси (θ, ϕ, α). Помимо этого, авторы предпочитают предсказывать размеры блока D, а не смещение T, потому что дисперсия оценки размеров, как правило, ниже (например, автомобили, как правило, имеют примерно одинаковый размер).

Для этого в модели используется регрессия, параметрами которой выстпуают па-

раметры поворота трехмерного блока, что в проекции будет изменять наложенность на двумерную рамку и соответвующее значение функции ошибки, которая будет описана позже. Количество возможных соответствий проекций трехмерных 8-ми вершин на 4 двумерные стороны рамки соответствует $8^4 = 4096$ возможностям. Но во множестве сценариев можно предположить, что объекты всегда находятся в вертикальном положении. В этом случае верх и низ параллелограма соответствуют только проекции верхних и нижних вершин 3D-рамки соответственно, что уменьшает количество соответствий до 1024. Кроме того, когда относительный крен (угол α) объекта близок к нулю, вертикальные координаты стороны параллелограма x_{min} и x_{max} могут соответствовать только проекциям точек с вертикальных сторон 3D-рамки. Аналогично, y_{min} и y_{max} могут соответствовать только точечным проекциям со сторон горизонтальной трехмерной рамки. Следовательно, каждая вертикальная сторона 2D-рамки обнаружения может соответствовать $[\pm d_x/2, ..., \pm d_z/2]$, а каждая горизонтальная сторона 2D-рамки соответствует $[..., \pm d_y/2, \pm d_z/2]$, остается $4^4 = 256$ возможных конфигураций. В наборе данных KITTI [10] оба угла наклона и крена объекта равны нулю, что еще больше уменьшает число конфигураций до 64

Рис. 2: Слева: обрезанное изображение проезжающей машины. Справа: изображение всей сцены. Как видно, автомобиль на обрезанных изображениях вращается, в то время как направление автомобиля является постоянным среди всех различных рядов.

Самой важной частью в описываемой статье является определение ориентации трехмерной охватывающей рамки. При перемещении одного и того же реального объекта в кадре некоторые модели могут ошибаться при вычислении ориентации трёхмерной охватывающей сетки, несмотря на то, что реальное направление объекта может не изменяться. На рисунке 2 демонстрируется упомянутый эффект. Подобные ошибки возникают в силу изменений, которые вносит проективное преобразование камеры во время съемки, поэтому необходимо минимизировать погрешность при определении направления трёхмерной рамки. В связи с этим рассматривается лишь угол поворота азимута для матрицы R, то есть сумму двух углов $\theta_{ray} + \theta_l$, что подробнее изображено на Рис.3.

В статье решение проблемы нахождения этого угла θ_l, которое может принимать значения в промежутке от 0 до 360 градусов, заключается в делении этого промежут-

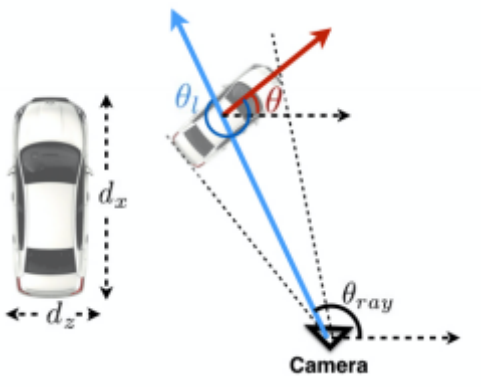

Рис. 3: Иллюстрация локальной ориентации θ_l и глобальной ориентации автомобиля θ. Локальная ориентация вычисляется относительно луча, проходящего через центр двумерной рамки. Центральный луч обозначен синей стрелкой. Обратите внимание, что центр рамки может не быть фактическим центром объекта. Ориентация автомобиля θ равна $\theta_{ray} + \theta_l$. Сеть обучена оценивать локальную ориентацию θl.

ка на условные n пересекающихся пронумерованных подотрезков $\Delta\theta_i$. Для каждого подотрезка сверточная нейронная сеть оценивает доверительную вероятность c_i того, что выходной угол находится внутри i-го подотрезка и поправку на остаточное вращение, которую необходимо применить к ориентации центрального луча этого подотрезка, чтобы получить выходной угол θ_l. Остаточное вращение представлено двумя числами, для синуса и косинуса угла. Это приводит к 3 выводам для каждого подотрезка i: $(c_i, cos(\Delta\theta_i), sin(\Delta\theta_i))$. Оценка этого параметра происходит путем применения функции потерь следующего вида:

$$L_\theta = L_{conf} + \omega \times L_{loc},$$

где

$$L_{conf_i} = \frac{e^{a_i}}{\sum_{k=1}^{N} e^{a_k}} \tag{3.2}$$

есть softmax, и на выводе получаем вероятности нахождения на каждом подотрезке, то есть параметров c_i, ω - гиперпараметр. L_{loc} высчитывается следующим образом:

$$L_{loc} = -\frac{1}{n_{\theta^*}} \sum \cos(\theta^* - c_i - \Delta\theta_i),$$

где n_{θ^*} является количеством подотрезков, которые покрывают истинный угол θ^*, а $\Delta\theta_i$ - это изменение, которое необходимо применить к центру подотрезка i.

Во время вывода выбирается ячейка с максимальной вероятностью, и окончательный результат вычисляется путем применения оценочного значения $\Delta\theta$ этого подотрезка к центру этого подотрезка. В результате необходимо оценить 3n параметров для n подотрезков. После подсчета всех параметров остается лишь посчитать решение системы, которая была описана выше, состоящая из четырех уравнений.

§ 3.2 Определение марки автомобиля и визуализация

Эксперименты были проведены на наборах данных KITTI [10] и Pascal 3D+ [11]. Набор данных KITTI имеет в общей сложности 7481 тренировочных образов. Мы обучаем объект-детектор MS-CNN [12] по созданию двумерной рамки, а затем оцениваем параллелепипед из двумерной рамки обнаружения, чьи оценки превышают пороговое значение. Для регрессии трехмерных параметров используется предварительно обученная сеть VGG [13] без ее слоев FC, и добавляется наш модуль 3D-рамки. Pascal3D+ - это набор данных состоящий из изображений из Pascal VOC и Imagenet для 12 различных категорий, которые снабжены позицией камеры. Изображения из обучающего набора Pascal и Imagenet используются для обучения, а оценка выполняется на проверочном наборе Pascal. В отличие от KITTI, внутренние параметры являются приблизительными, и поэтому невозможно восстановить истинные размеры физического объекта.

После расчета вершин трехмерной рамки по представленным вершинам двумерной рамки вырезается картинка из исходного изображения. Так как двумерная рамка слишком плотно прилигает к объекту, делается небольшой отступ в сторону увеличения относительно центра. Объясняется это тем, что на обучении для второй сети в тренировачной базе данных использовались картинки хорошего качества, где объект изображен с четкими границами. Модель работает на сверточной нейронной сети и высчитывает классификацию среди 196 моделей автомобилей, находящихся в базе данных, предложенных стэнфордским университетом. Вывод данной части программы добавляется в общий вывод модели.

§ 3.3 Подсчет центра нижней грани параллелепипеда

В текущей версии модели решается упрощенная задача нахождения центра тяжести, как центра нижней грани параллелепипеда, поэтому в задаче игнорируется вывод модели и марки автомобиля. В будущем это планируется использовать для уточнения координаты центра тяжести.

Остается посчитать центр нижней грани. При проекции вершин нижней грани параллелепипеда на изображении получается параллелограмм. Проекция цетнра нижней грани является центром этого параллелограмма и высчитывается как точка пересечения диагоналей.

§ 4 Заключение

В данной работе применена модель, которая использует вычисления в реальном времени и может найти применение на практике. Модель можно запустить по адресу `https://github.com/TimurKinazar/3DBoundingBox`

Но также в модели имеются некоторые проблемы, которые не удалось решить в рамках данной курсовой работы. В частности, если снимать покадрово видео и применять данный метод, то он высчитывает трехмерные рамки с видимой человеческим взглядом разрывом, то есть при гладком изменении покадрово объекта довольно резко меняются вершины параллелепипеда, что влияет на оценку нахождения центра нижней грани. Также вывод второй нейронной сети сильно отличается от того, что в реальности изображено в плане вычисления модели автомобиля. Проблема заключается в низком качестве вырезаемого изображения, на котором находится только изучаемый объект, и в малом количестве моделей в обучающем датасете.

В будущем планируется добавить сеть, высчитывающую дальность объекта относительно камеры для корректировки трехмерной рамки, а также подключение базы автомобильных номеров для более четкого определения модели и марки автомобиля, использования этого параметра, как дополнительного известного числового параметра, например, ширины и длины. Основной идеей является расчет размеров и ориентации объекта восстановлением трехмерной модели из соответствующего класса из набора данных Pascal3D+.

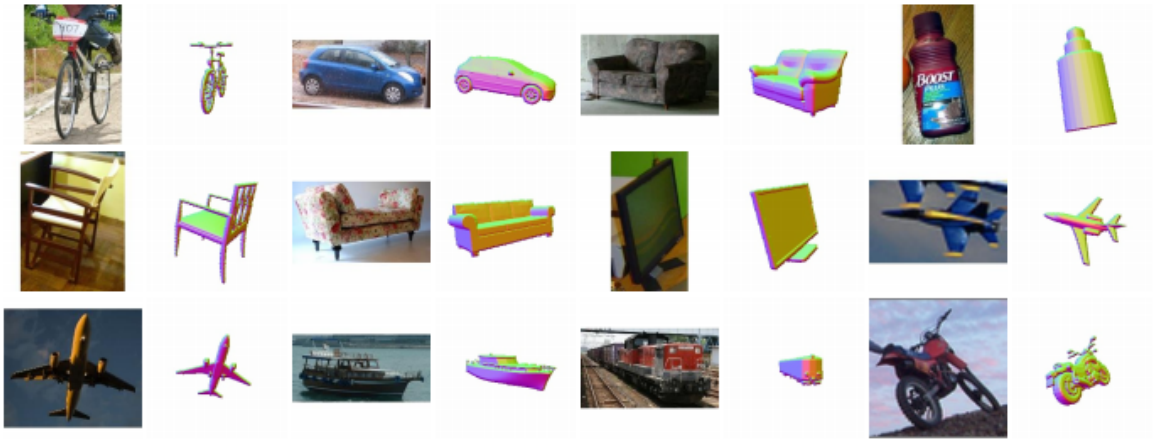

Рис. 4: Визуализация моделей в наборе данных Pascal3D+.

Отдельную благодарность за помощь в написании данной работы хотелось бы выразить аспиранту механико-математического факультета кафедры математической теории интеллектуальных систем Ронжину Дмитрию Владимировичу, а также научному руководителю доценту кафедры Математической теории интеллектуальных систем, кандидат физико-математических наук Часовских Анатолию Александровичу.

§ 5 Пример вывода сети

Рис. 5: Пример вывода 1.

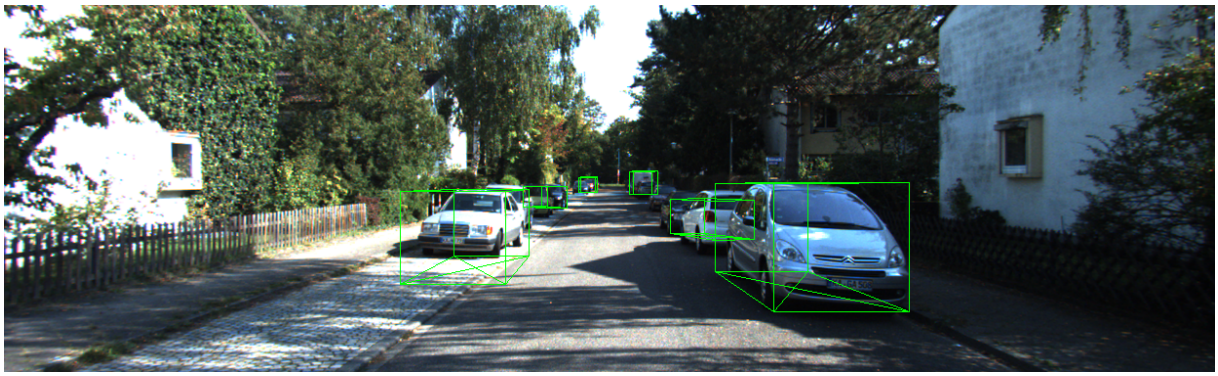

Рис. 6: Пример вывода 3.

```
Estimated pose: 283 185 345 211
The centre of mass coordinates is [ 329 , 212 ]
Lamborghini Aventador Coupe 2012


Estimated pose: 366 182 408 202
The centre of mass coordinates is [ 394 , 203 ]
Aston Martin V8 Vantage Coupe 2012


Estimated pose: 549 179 567 185
The centre of mass coordinates is [ 560 , 187 ]
Geo Metro Convertible 1993


Got 3 poses in 3.296 seconds
-------------
```

Рис. 7: Пример вывода текстовой составляющей.

Литература

1. Chenge Li, Gregory Dobler, Xin Feng, Yao Wang *TrackNet: Simultaneous Object Detection and Tracking and Its Application in Traffic Video Analysis* — In Computer Vision and Pattern Recognition 2019

2. Xiaoliang Wang, Peng Cheng, Xinchuan Liu, Benedict Uzochukwu *Focal Loss Dense Detector for Vehicle Surveillance* — Computer Vision and Pattern Recognition 2018

3. Ронжин Д.В. *Распознавание динамики особых точек в видеоряде* — Журнал: Интеллектуальные системы. Теория и приложения Том: 18 Номер: 2 2014г. 267-276стр

4. Florian Chabot, Mohamed Chaouch, Jaonary Rabarisoa *Deep MANTA: A Coarse-to-fine Many-Task Network for joint 2D and 3D vehicle analysis from monocular image* — In CVPR 2017

5. Jason Ku, Alex D. Pon, Steven L. Waslander *Monocular 3D Object Detection Leveraging Accurate Proposals and Shape Reconstruction* — In CVPR 2019

6. Arsalan Mousavian, Dragomir Anguelov, John Flynn, Jana Kosecka *3D Bounding Box Estimation Using Deep Learning and Geometry.*— In IEEE CVPR 2017

7. P. L. A. Geiger and R. Urtasun. *Are we ready for autonomous driving? the KITTI vision benchmark suite.* In CVPR, 2012.

8. Soroush. *PyTorch implementation for this paper.* — https://github.com/skhadem/3D-BoundingBox

9. Nithiroj T. *Car Recognition* — https://github.com/nithiroj/car-recognition

10. A Benchmark for 3D Object Detection in the Wild — http://cvgl.stanford.edu/projects/pascal3d.html

11. Y. Xiang, R. Mottaghi, and S. Savarase. *Beyond pascal: A benchmark for 3d object detection in the wild.* — In WACV, 2014

12. Z. Cai, Q. Fan, R. Feris, and N. Vasconcelos. *A unified multi-scale deep convolutional neural network for fast object detection.* — In ECCV, 2016

13. K. Simonyan and A. Zisserman. *Very deep convolutional networks for large-scale image recognition.* — CoRR, abs/1409.1556, 2014

14. Stanford Dataset for Car Recognition — https://ai.stanford.edu/ jkrause/cars/

15. Vladimir V. Kniaz Peter V. Moshkantsev Vladimir A. Mizginov *Deep Learning a Single Photo Voxel Model Prediction from Real and Synthetic Images* In XXI International Conference on Neuroinformatics, October 7-11, 2019, Dolgoprudny, Moscow region, Russia

16. M.Zeeshan Zia, Michael Stark, Konrad Schindler *Towards Scene Understanding with Detailed 3D Object Representations* — International Journal of Computer Vision 2014