ОЦЕНКА ГЛУБОКИХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ СИСТЕМЫ ЛОКАЛИЗАЦИИ БАГАЖНЫХ БИРОК В АЭРОПОРТУ

Ивлиев Е.А., Обухов П.С.

Донской государственный технический университет, Ростов-на-Дону, e-mail: 123ivliev123@mail.ru

В этой статье анализируется современные подходы к решению задачи обнаружения объектов на основе нейронных сетей с такими мета-архитектурами как: Faster R-CNN, R-FCN и SSD в сочетании с различными нейронными сетями извлекающими признаки с архитектурами: Resnet V1 50, Resnet V1 101, Inception V2, Inception Resnet V2 и Mobilenet V1. Мы стремимся исследовать свойства данных моделей обнаружения объектов, которые модифицированы и специально адаптированы к проблемной области детектирование багажных бирок в аэропорту для дальнейшего извлечения информации о коде аэропорта IATA. Оценка и сравнение этих моделей включают ключевые метрики, такие как точность, потребление памяти, время работы, количество операций с плавающей запятой, количество параметров модели. Наши результаты показывают, что Faster R-CNN Inception Resnet V2 имеют лучшую точность, в то время как R-FCN Resnet 101 предлагает лучший компромисс между точностью и временем обработки. SSD Mobilenet V2 заслуживает особого упоминания, так как является самой быстрой и легкой моделью с точки зрения потребления памяти, что делает его оптимальным выбором для развертывания в мобильных и встраиваемых устройствах.

Ключевые слова: локализация объектов, глубокие нейронные сети, мета-архитектура, извлечение признаков, TensorFlow

ASSESSMENT OF DEEP NEURAL NETWORKS FOR LOCALIZATION SYSTEM OF BAGGAGE TAGS AT AIRPORT

Ivliev E.A., Obukhov P.S.

Don state technical University, Rostov-on-Don, e-mail: 123ivliev123@mail.ru

This paper analyzes modern approaches to solving the problem of detecting objects based on neural networks with such meta-architectures as: Faster R-CNN, R-FCN and SSD in combination with various neural networks extracting features with architectures: Resnet V1 50, Resnet V1 101, Inception V2, Inception Resnet V2 and Mobilnet V1. We aim to investigate the properties of these object detection models, which are modified and specially adapted to the problem area of baggage tag detection at the airport to further retrieve IATA airport code information. Evaluation and comparison of these models include key metrics such as accuracy, memory consumption, running time, FLOPS, number of model parameters. Our results suggest that Faster R-CNN Inception Resnet V2 have better accuracy, while R-FCN Resnet 101 offers a better compromise between accuracy and execution time. SSD Mobilenet V2 deserves special mention, as it is the fastest and easiest model in terms of memory consumption, which makes it the optimal choice for deployment in mobile and embedded devices.

Keywords: localization of objects, deep neural networks, meta-architecture, feature extraction, TensorFlow

Проблема идентификации объектов в видеопотоке является одной из наиболее востребованных в сфере технического зрения. На ее основе решается множество прикладных задач. В данной работе в качестве объектов рассматриваются багажные бирки в сортировочной зоне аэропорта.

Актуальность данной работы обусловлена тем, что сотрудники сортировочных помещений аэропорта лишены возможности простой идентификации багажных бирок с помощью сканеров штрих-кодов потому.

В последние годы большинство современных алгоритмов обнаружения объектов, таких как Faster R-CNN [1], R-FCN [2] и SSD [3], использовали сверточные нейронные сети (CNN) и могут быть развернуты в мобильных устройствах и потребительских продуктах. Для того чтобы определить, какой детектор лучше всего подходит для определенного применения, важны не только стандартные метрики точности, такие как средняя точность, но и другие факторы, такие как потребление памяти и время работы, также играют критическую роль [4].

Поскольку многие из ведущих современных подходов к обнаружению объектов сошлись на общей методологии, которая состоит из одного CNN, который использует прогнозы в стиле скользящего окна и обучен со смешанной целью регрессии и классификации, авторы реализуют мета-архитектуры Faster R-CNN, R-FCN и SSD в сочетании с различными архитектурами, извлекающими признаков, для того чтобы сравнивать большое количество систем обнаружения унифицированным образом.

В этой статье анализируются и сравниваются семи моделей CNN для обнаружения объектов, которые ранее были разработаны и предварительно обучены. Оцененные модели обнаружения представляют собой комбинации мета-архитектур (Faster R-CNN, R-FCN и SSD) и экстракторов признаков (Resnet V1 50, Resnet

V1 101 [5], Inception V2 [6], Inception Resnet V2 [7] и Mobilenet V1 [8]).

1. Обзор мета-архитектуры для детектирования объектов

1.1. Faster R-CNN. Нейросеть детектирования объектов, называемая Faster R-CNN, состоит из двух модулей. Первый модуль представляет собой глубокую полностью сверточную сеть, которая определяет регионы предполагаемых объектов Region Proposal Networks (RPN), а второй модуль представляет собой детектор Fast RCNN который использует ранее определенные регионы для классификации объектов внутри данных регионов. Вся система представляет собой единую унифицированную сеть для обнаружения объектов.

Чтобы предсказывать регионы, RPN использует карту признаков последнего сверточного слоя, с которого значения передаются в два параллельных полносвязанных слоя: слой регрессии (reg layer) и слой классификации (cls layer).

В каждом месте карты признаков сеть одновременно предсказывает несколько предложений регионов, где число максимально возможных предложений для каждого места обозначается как k. k предложений параметризованный относительно k ссылочных блоков, которые называются якорями. Якорь центрирован на скользящем окне, и связан с масштабом и соотношением сторон. Используется 3 масштаба и 3 соотношения сторон, что дает k = 9 якорей на каждой позиции скольжения [1].

Во время работы RPN слой классификации по каждому якорю присваивает метку двоичного класса (является объектом или нет). Положительная метка присваивается двум видам якорей:

– якорь с наивысшей оценкой перекрытия по отношению к истинному значению ограничивающей рамки;

– якорь, который имеет оценку перекрытия выше 0,7 с любым истинным значением.

Отрицательная метка (не является объектом) присваивается якорю, если его оценка перекрытия ниже 0,3 для любого истинного значения. Остальные якоря не вносят вклад в обучение.

Далее Faster R-CNN, используя полученные координаты из слоя регрессии, подает их на RoiPooling слой, который выделяет области интереса исходного изображения и подает каждую из них нескольким полносвязным слоям для классификации области изображения и для уточнения ее координат.

Для экспериментов количество предложений по регионам, которые должны быть отправлены в классификатор ограничивающих рамок, устанавливается равным 300. Кроме того, каждый экстрактор признаков обучают изображениям, масштабированным до 600 пикселей, используя SGD оптимизатор [9], размер партии равен 1. Начальная скорость обучения устанавливается в 0,0003 и вручную уменьшается в 10 раз: после 900 000 итераций и 1 200 000 итераций.

1.2. R-FCN. Region-based Fully Convolutional Networks (R-FCN) используют архитектуру Faster R-CNN, но только со сверточными нейронными сетями. В отличии от Faster R-CNN обрезка областей не происходит на выходе сети прогнозирования регионов, вместо этого к входу первой сети добавляется сверточный слой для дополнительного извлечения признаков и обрезка областей производится из последнего сверточного слоя. Далее происходит классификация с помощью всего лишь одного или двух сверточных слоев нейронов. Такой подход позволил достичь точности сравнимой с Faster R-CNN при более быстром времени работы [2].

Конфигурация обучения, а также настройка параметров R-FCN такие же как у Faster R-CNN.

1.3. SSD. По сравнению с архитектурами Faster R-CNN и R-FCN, SSD сводит все вычисления в единую сверточную нейронную сеть с выводом ограничивающих рамок и классов объектов. На выход этой нейросети формируется несколько тысяч различных прогнозов для возможных регионов расположения объектов разной формы на разных масштабах, затем с помощью подавления немаксимумов (Non-Maximum Suppression) происходит выбор нескольких наиболее вероятных областей. Такая единая структура, одновременно с учетом различных масштабов изображения обеспечила методу SSD наиболее высокие показатели по скорости и качеству обнаружения объектов по сравнению с остальными современными подходами [3].

Для экспериментов, в отличие от Faster R-CNN и R-FCN, модели SSD обучаются с использованием оптимизатора RMSprop и размером партии 16. Базовая скорость обучения устанавливается равной 0,004 и экспоненциально затухает на коэффициент 0,95 для каждых 800000 итераций. Что касается размеров входного изображения, они имеют фиксированную форму 300×300 пикселей.

2. Проведение эксперимента. Наша экспериментальная установка состоит из трех мета-архитектур (Faster R-CNN, R-FCN и SSD) и шести сверточных нейросетей извлекающих признаки (Resnet V1 50, Resnet V1 101, Inception V2, Inception Resnet V2 и Mobilenet V1).

Из-за временных ограничений и вычислительных затрат во всех экспериментах, представленных в данной статье, используются общедоступные модели обнаружения объектов, которые были предварительно подготовлены на базе набора данных Microsoft COCO [10]. Все предварительно подготовленные модели, которые используются в нашей экспериментальной установке доступны в официальном хранилище Tensorflow Object Detection API. Комбинации мета-архитектур и архитектур извлекающих признаки, исследованные в этой работе, представлены в табл. 1. Можно заметить, что не все возможные комбинации были исследованы. Причина в том, что каждая нейросеть извлекающая признаки должна быть адаптирована для использования в мета-архитектуре. Эти не тривиальные корректировки требуют большого количества экспериментов и недель тренировок, и, следовательно, были выбраны только предварительно подготовленные комбинации.

Таблица 1

Комбинации мета-архитектур для детектирования объектов и архитектур для извлечения признаков

	Faster R-CNN	R-FCN	SSD
Resnet V1 50	✓		
Resnet V1 101	✓	✓	
Inception V2	✓		✓
Inception Resnet V2	✓		
Mobilenet V1			✓

Для обучения нейросети была создана обучающая выборка состоящая из 500 изображений с багажными бирками. Аннотация данных выполнялась программой LabelImg, с помощью которой выделяются границы интересуемого объекта и указывается класс к которому принадлежит данный объект.

3. Анализ результатов. В этом разделе представлены результаты экспериментов с детектором багажных бирок в аэропорту. Анализ каждого из этих экспериментов включает в себя множество измерений, таких как точность, количество параметров, операции с плавающей запятой (FLOP), потребление памяти и время обработки. Модели обучались и оцениваются на компьютере с процессором AMD Ryzen 7 1700, 24 ГБ оперативной памяти и дискретным графическим процессором NVIDIA GeForce GTX 1060, который имеет 1280 CUDA ядер и 6 ГБ памяти.

Для оценки эффективности работы детектора багажных бирок и штрих-кодов, как ориентира для поиска информации кода аэропорта IATA, используется такие метрики как мера пересечения предсказанных и истинных ограничивающих рамок, содержащих багажную бирку (Intersection, I), полноту (Recall, R) и точность (Precision, P) обнаружения объекта [11].

Мера пересечения предсказанных и истинных ограничивающих рамок I (1) показывает, насколько точно сверточная нейросеть предсказала координаты ограничивающей рамки относительно истинной разметки.

$$I = \frac{S_I}{S_f + S_{gt} - S_I}, \qquad (1)$$

где S_I – площадь пересечения предсказанной и истинной ограничивающей рамки, S_f – площадь предсказанной ограничивающей рамки, S_{gt} – площадь истинной ограничивающей рамки.

Полнота R (2) показывает чувствительность алгоритма к ошибкам 2-го рода, то есть, пропускам, и равна отношению количества правильно предсказанных объектов к общему количеству этих объектов в истинной разметке.

$$R = \frac{tp}{tp + fn}, \qquad (2)$$

где tp – истинно-положительные – те объекты, которые мы ожидали увидеть и получили на выходе, fn – ложно-отрицательные объекты которые мы ожидали увидеть, но алгоритм их не определил.

Точность P (3) показывает чувствительность алгоритма к ошибкам 1-го рода, то есть, ложным срабатываниям и равна отношению количества правильно предсказанных объектов, к общему количеству предсказанных алгоритмом ограничивающих рамок.

$$P = \frac{tp}{tp + fp}, \qquad (3)$$

где fp – ложно-положительные – объекты, которых быть на выходе не должно, но алгоритм их ошибочно вернул на выходе [11].

Подробные результаты точности, полноты и меры пересечения представлены в табл. 2. В табл. 3 представлен список моделей отсортированных по точности с такими характеристиками, как FPS, память, количество операций в секунду и количество параметров каждой модели.

Для нашей системы по детектированию багажных бирок время выполнения является критическим фактором. Точность достигнутая каждой конфигурацией модели,

вместе с ее временем обработки представлена на рис. 1. Наблюдаются три группы. Первая группа состоит из самых быстрых моделей с мета-архитектурой SSD, которые не выполняют формирование региональных предложений. SSD Mobilenet является самой быстрой из всех моделей, со временем выполнения обработки одного изображения 23,61 мс. (42 кадра в секунду), хотя его точность немного хуже, чем у SSD Inception V2. Вторая группа состоит из Faster R-CNN с упрошенными нейросетями извлекающими признаки и R-FCN Resnet 101. Эти модели более точны и требуют приблизительно 150 мс на изображение в среднем. На самом деле, точности, полученные R-FCN и Faster R-CNN, когда извлекающим элементом является сеть Resnet 101, очень близки к модели Faster R-CNN Inception Resnet V2 (третья группа), точность которой составляет 84,41%. Однако на сегодняшний день это самая медленная модель из-за времени ее обработки, которое составляет 641 мс. Следовательно, модель R-FCN Resnet 101 обеспечивает наилучший баланс между точностью и скоростью среди изученных конфигураций модели, так как ее точность достигает 82,67%, а время обработки одного изображения занимает 108,57 мс на изображение (9,21 fps).

На рис. 2 представлена зависимость количества операций в секунду (FLOPS) от времени обработки одного изображения. Число FLOPS, вычисленное каждой моделью, является не зависящим от платформы измерением. Анализируя эти данные можно сказать, что использование более плотных блоков в нейросетях с архитектурой ResNet приводит к увеличению FLOPS и времени вычислений как для Faster R-CNN, так и для R-FCN детекторов. Следует отметить, что SSD Mobilenet – это модель с наименьшим количеством FLOPS и наименьшим временем работы.

Таблица 2

Результаты точности детектирования багажной бирки, полученные с помощью каждой модели

Модели	Мера пересечения (I),%	Точность (P),%	Полнота (R),%
Faster R-CNN Resnet 50	83,26	82,3	85,71
Faster R-CNN Resnet 101	87,74	78,65	93,88
Faster R-CNN Inception V2	81,23	79,45	81,63
Faster R-CNN Inception Resnet V2	91,68	84,41	93,88
R-FCN Resnet 101	87,37	82,67	93,54
SSD Inception V2	82,75	68,34	60,41
SSD Mobilenet V1	80,51	65,21	58,03

Таблица 3

Характеристики моделей, отсортированные по точности

Модели	Точность (P),%	FPS, 1/c	Память, MB	Количество операций в секунду (FLOPS * 10^9)	Количество параметров (*10^6)
Faster R-CNN Inception Resnet V2	84,41	1,56	18250,45	1837,54	59,41
R-FCN Resnet 101	82,67	9,21	3509,75	269,9	64,59
Faster R-CNN Resnet 50	82,3	6,81	5256,45	533,58	43,34
Faster R-CNN Inception V2	79,45	13,12	2175,21	120,62	12,89
Faster R-CNN Resnet 101	78,65	6,11	6134,71	625,78	62,38
SSD Inception V2	68,34	31,42	284,51	7,59	13,47
SSD Mobilenet V1	65,21	42,34	94,7	2,3	5,57

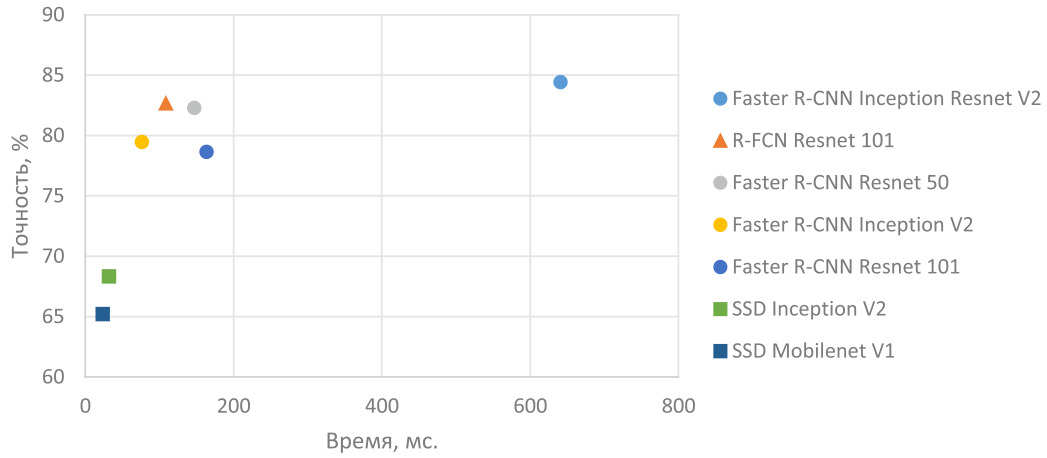

Рис. 1. Зависимость точность детектирования от времени обработки изображения

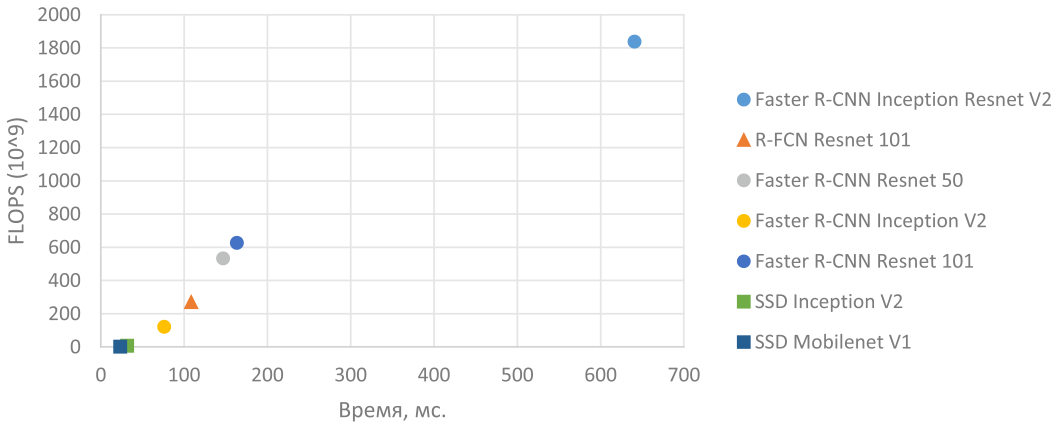

Рис. 2. Зависимость количество операций в секунду от времени обработки изображения

Анализируя количество параметров, которые каждая нейронная сеть должна изучить (веса и смещение), было выяснено, что они не связаны напрямую с временем обработки, рис. 3. Можно видеть, что модели, в которых нейросеть извлекающая признаки является Resnet 101, содержат количество параметров, соизмеримое с моделью Faster R-CNN Inception Resnet V2, однако время обработки изображения намного ниже. Модели SSD Mobilenet, SSD Inception V2 и Faster R-CNN Inception V2 имеют наименьшее время обработки, но и наименьшее количество параметров.

Потребление памяти также является критическим фактором. Это помогает принимать решения, такие как, может ли определенная модель быть обучена на одном GPU или необходимо использовать кластер этих вычислительных блоков, и решать, может ли определенная архитектура нейронной сети быть развернута в мобильных и встраиваемых устройствах. На рис. 4 представлено общее использование памяти в зависимости от времени обработки изображения каждой моделью. Существует высокая линейная корреляция между временем выполнения и большими и более мощными экстракторами функций, которые требуют гораздо больше памяти. Модели, основанные на основе архитектуры ResNet, занимают верхние позиции с точки зрения использования памяти, в то время как модели SSD Mobilenet и SSD Inception V2 являются самыми дешевыми в том, что они требуют 94,70 МБ и 284,51 МБ соответственно.

Наконец, на рис. 5 изображена Лепестковая диаграмма, оси которой представляют пять измеренных характеристик, которые описывались выше: точность, время обработки, количество операций в секунду (FLOPS), параметры и количество потребляемой памяти. Минимальное значение каждого показателя рассматривалось

как лучшее, за исключением точности, где максимальное значение принималось как лучшее. Кроме того, для каждого фактора все значения были преобразованы в диапазон [0, 10]. Следует иметь в виду, что точность, время работы и потребление памяти являются наиболее критическими факторами. Следовательно, мы наблюдаем, что лучшими общими моделями являются R-FCN Resnet 101 и Faster R-CNN Inception V2.

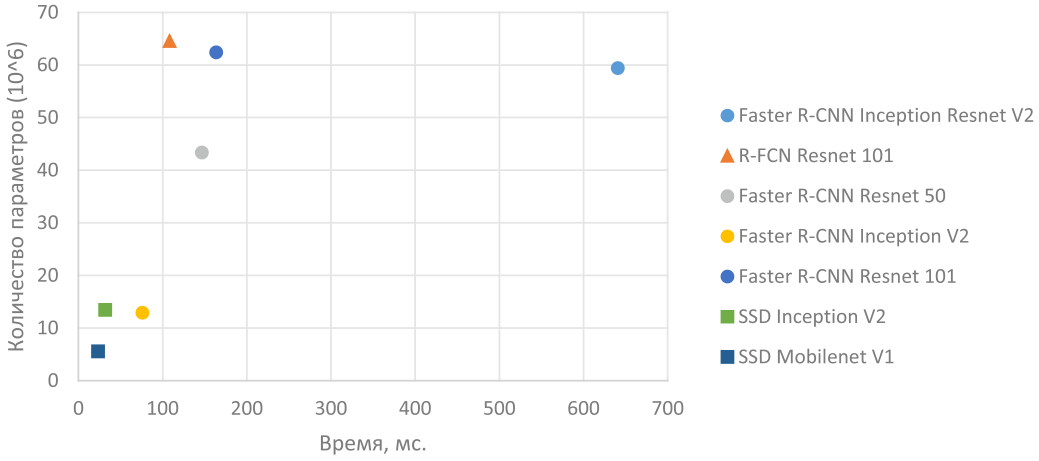

Рис. 3. Зависимость количество параметров от времени обработки изображения

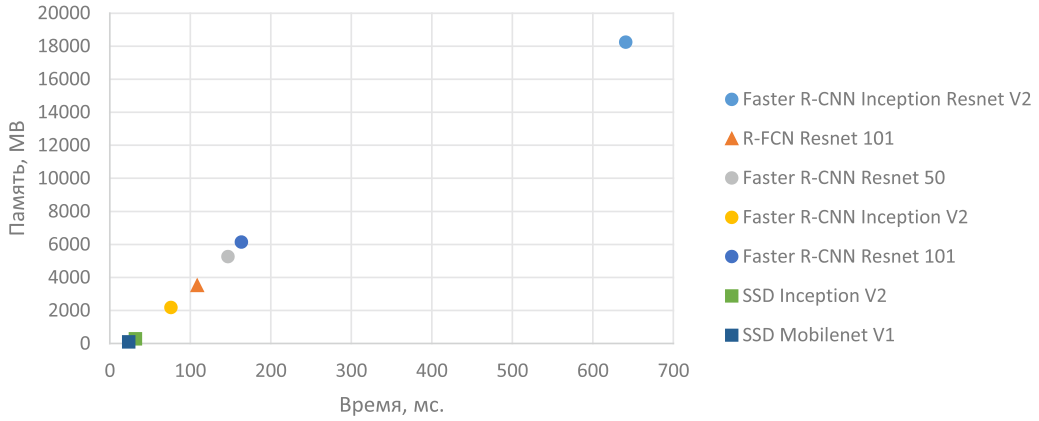

Рис. 4. Зависимость потребляемой памяти от времени обработки изображения

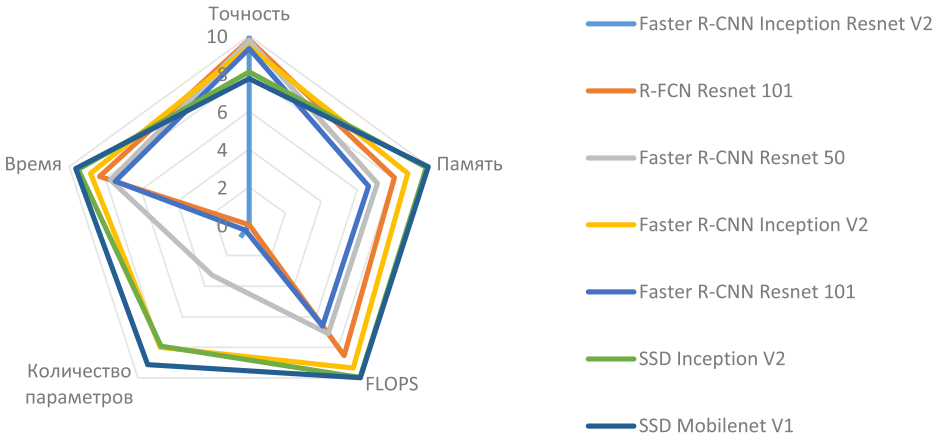

Рис. 5. Лепестковая диаграмма анализа моделей для детектирования багажных бирок по всем параметрам

Заключение

В этой статье представлено экспериментальное сравнение семи детекторов багажной бирки на основе глубоких нейронных сетей. Проанализированы основные аспекты этих детекторов, такие как точность, скорость, потребление памяти, количество операций с плавающей запятой и количество обучаемых параметров в CNN.

Было обнаружено, что Faster R-CNN Inception Resnet V2 имеет самую высокую точность (84,41%), в то время как R-FCN Resnet 101 имеет лучший компромисс между точностью (82,67%) и временем обработки (108,57 мс на изображение). Большого внимания заслуживает SSD Mobilenet, которая является самой быстрой моделью из всех детекторов, а также наименее требовательной с точки зрения потребления памяти. Эти ключевые факторы делают SSD Mobilenet оптимальным выбором для развертывания в мобильных и встраиваемых устройствах. Также следует отметить, что только модели SSD достигают более 30 FPS с помощью NVIDIA GTX 1060, что позволяет их использовать в реальном времени.

Список литературы

1. Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. Neural Information Processing Systems. 2015. vol. 39. P. 1137–1149.

2. Dai J., Li Y., He K., Sun J., R-fcn: Object detection via region-based fully convolutional networks. Neural Information Processing Systems. 2016. vol. 1. P. 379–387.

3. Liu W., Anguelov D., Erhan D., Szegedy C., Reed S., Fu C.Y., Berg A.C. SSD: Single shot multibox detector. European Conference on Computer Vision. 2016. vol. 1. P. 21–37. DOI: 10.1007/978-3-319-46448-0_2.

4. Huang J., Rathod V., Sun C., Zhu M., Korattikara A., Fathi A., Fischer I., Wojna Z., Song Y., Guadarrama S., Speed/accuracy trade-offs for modern convolutional object detectors IEEE Conference on Computer Vision and Pattern Recognition. 2017 vol. 1. P. 7310-7319. DOI: 10.1109/CVPR.2017.351.

5. He K., Zhang X., Ren S., Sun J. Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. vol. 1. P. 770–778. DOI:10.1109/CVPR.2016.90.

6. Ioffe S., Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. Proceedings of Machine Learning Research. 2015. vol. 37. P. 448–456.

7. Szegedy C., Ioffe S., Vanhoucke V. Inception-v4, inception-resnet and the impact of residual connections on learning. AAAI Conference on Artificial Intelligence. 2017. vol. 1. P. 4278–4284.

8. Howard A.G., Zhu M., Chen B., Kalenichenko D., Wang W., Weyand T., Andreetto M., Adam H., Mobilenets: Efficient convolutional neural networks for mobile vision applications. 2018 cite arXiv:1602.07261.

9. Qian N. On the momentum term in gradient descent learning algorithms. 1999. DOI: 10.1016/ S0893-6080(98)00116-6.

10. Lin, T.Y. Microsoft COCO: Common objects in context / M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick // In ECCV, 2014. DOI: 10.1007/978-3-319-10602-1_48.

11. Чуйков Р.Ю., Юдин Д.А., Обнаружение транспортных средств на изображениях загородных шоссе на основе метода Single Shot Multibox Detector // Научный результат. Информационные технологии. 2017. № 4. С. 50–58.