

Раздел IV. Связь, навигация и наведение

УДК 004.93'12

DOI 10.18522/2311-3103-2020-1-188-199

В.А. Бондаренко, В.А. Павлова, В.А. Тупиков, Н.Г. Холод

НЕЙРОСЕТЕВОЙ АЛГОРИТМ ПОЛНОКАДРОВОГО РАСПОЗНАВАНИЯ НАДВОДНЫХ ОБЪЕКТОВ В РЕАЛЬНОМ ВРЕМЕНИ

Исследованы современные нейросетевые архитектуры для осуществления автоматического обнаружения и распознавания надводных объектов и препятствий заданных классов по всей области изображения, применимые к выполнению в реальном или условно реальном времени для задач оптоэлектронной системы технического зрения с целью автоматизации и повышения безопасности гражданского судоходства. Дана формальная постановка задачи автоматического обнаружения объектов на изображениях. Проанализирован текущий научно-исследовательский задел в области алгоритмов детектирования объектов на изображениях, основанных на искусственных свёрточных нейронных сетях, произведено их сравнение и сделан обоснованный выбор в пользу наиболее эффективной нейросетевой архитектуры по соотношению вычислительной сложности и точности распознавания. Исследованы имеющиеся в открытом доступе базы данных образов надводных объектов, подходящие для применения при обучении алгоритмов с использованием искусственных нейронных сетей. Сделан вывод о недостаточности имеющихся данных для обучения нейросетевых алгоритмов, в результате чего авторами выполнен самостоятельный сбор исследовательских изображений и видеопоследовательностей, произведена подготовка и разметка собранных данных, содержащих надводные объекты и иные препятствия, представляющие навигационную опасность для судов. На основе выбранной архитектуры разработан новый нейросетевой алгоритм автоматического обнаружения и распознавания надводных объектов, выполняемого по всей области изображения (т.е. полнокадрового), выполнено обучение искусственной нейронной сети по подготовленной базе данных образов типовых объектов. Полученный алгоритм протестирован авторами на валидационном наборе данных, произведена оценка качества его работы с помощью различных метрик, а также оценена производительность алгоритма. Сделаны выводы о необходимости расширения собранной базы данных образов типовых объектов, предложены дальнейшие шаги по повышению точности разработанного программно-алгоритмического комплекса и его внедрению в состав перспективной судовой оптоэлектронной системы технического зрения.

Безэкипажное судоходство; автоматическое обнаружение и распознавание; искусственная нейронная сеть; базы данных образов.

V.A. Bondarenko, V.A. Pavlova, V.A. Tupikov, N.G. Holod

REALTIME NEURAL NETWORK ALGORITHM FOR FULL-FRAME MARINE SURFACE OBJECTS RECOGNITION

The article explores modern neural network architectures for the automatic detection and recognition of marine surface objects and obstacles of given classes throughout the full image area, applicable for execution in real or near real time on an optoelectronic vision system to automate and improve the safety of civil marine navigation. A formal statement of the problem of automatic detection of objects on images is given. The state-of-the-art algorithms for detecting objects in images based on use of artificial convolutional neural networks were reviewed, their comparison was made and a reasonable choice was made in favor of the most efficient neural

network architecture in terms of computational complexity to recognition accuracy. The subject area is studied, as well as publicly available databases of surface objects suitable for use in the training of algorithms using artificial neural networks. The article concluded that there is insufficient labeled data for training neural network algorithms, as a result of which the authors independently collected research images and video sequences, prepared and labeled the collected data containing surface marine objects and other obstacles that represent a navigation hazard for ships. Based on the selected neural network architecture, a new neural network algorithm for automatic full-frame detection and recognition of surface objects was developed, and an artificial neural network was trained using the prepared database of images of typical objects. The resulting algorithm was tested by the authors on a validation data set, the quality of its work was estimated using various metrics, and the algorithm's performance was measured. Conclusions are made about the necessity to expand the collected database of images of typical marine objects, further steps are proposed to improve the accuracy of the developed software and algorithmic complex and its implementation to be used in a marine optoelectronic machine vision system for automation and improving the safety of civil navigation.

Unmanned marine navigation; automatic detection and recognition; artificial neural network; image database.

Введение. В настоящее время активно развивается отрасль морских перевозок, в частности за счет увеличения грузооборота во многих значимых морских торговых путях. Дальнейшему расширению морских грузоперевозок может способствовать снижение издержек путем уменьшения численности экипажа судов.

Поэтому становятся особо востребованными системы, обеспечивающие возможность автоматического управления судами и возможность дистанционного управления судами из берегового центра. Важной частью подобных систем является система, позволяющая передавать информацию об окружающих морское судно объектах в систему автоматического управления движением судна.

Задача создания такой системы является нетривиальной, поскольку, во-первых, подобная система должна иметь высокую скорость работы и не допускать задержек, а во-вторых, должна обеспечивать высокую надежность распознавания объектов, представляющих навигационную опасность для морских судов.

Среди алгоритмов детектирования объектов на изображении особо выделяются искусственные свёрточные нейронные сети (ИСНС) [1, 2], которые на данный момент являются наилучшими моделями для решения подобного типа задач с точки зрения точности распознавания. Кроме того, нейронные сети допускают эффективную реализацию с использованием систем массового параллелизма на GPU и FPGA.

Данная работа посвящена построению прототипа детектора надводных объектов на изображениях с помощью искусственной сверточной нейронной сети распознавания по всей области изображения (полнокадрового).

Формальная постановка задачи. Пусть имеется изображение I . Алгоритм распознавания должен, получив на вход данное изображение, предоставить на выходе n последовательностей вещественных чисел

$$(x_1^j, y_1^j, x_2^j, y_2^j, c_1^j, \dots, c_i^j, \dots, c_k^j),$$

где n – количество обнаруженных объектов на изображении, $1 \leq j \leq n$ – номер текущего распознанного объекта, x_1^j, y_1^j – координаты левой верхней вершины прямоугольника, ограничивающего объект (bounding box), x_2^j, y_2^j – координаты его правой нижней вершины, k – количество классов объектов, которые способна обнаруживать нейронная сеть, c_i^j – оценка вероятности того, что j -ый объект принадлежит классу с номером i .

В данной работе в качестве алгоритмов рассматриваются только искусственные свёрточные нейронные сети. Для того, чтобы данные алгоритмы смогли работать, необходимо обучить их на некоторой обучающей выборке X , для каждого

изображения $I \in X$ известны координаты ограничивающих прямоугольников для всех объектов интереса, присутствующих на нем. Создание данной выборки является одним из самых важных и сложных вызовов, встающих при обучении искусственных нейронных сетей. Процесс создания такой выборки для данной задачи будет подробно рассмотрен ниже.

Обзор литературы. Существующие архитектуры свёрточных нейронных сетей для обнаружения объектов на изображениях можно разделить на две категории: одноэтапные (one-stage) и двухэтапные (two-stage).

Двухэтапные нейросетевые алгоритмы детектирования объектов на изображении включают в себя два этапа.

Первый этап заключается в поиске подозрительных зон на изображении, т. е. зон, возможно содержащих интересующие нас объекты. Данный этап может происходить без использования нейронных сетей с помощью информации о контрасте, по ключевым точкам или перебирая все возможные положения объекта с помощью процедуры selective search. Однако зоны, полученные вышеперечисленными методами, зачастую имеют очень низкое качество в том смысле, что много областей либо содержат слишком большое количество фона, либо содержат только небольшую часть объекта, либо содержат более одного объекта. Поэтому более предпочтительно на данном этапе использовать свёрточные нейронные сети, не содержащие полносвязных слоев (fully convolutional).

Второй этап заключается в классификации зон, найденных на первом этапе. На данном этапе для классификации всегда используются обычные свёрточные искусственные нейронные сети.

Среди двухэтапных алгоритмов детектирования объектов на изображении особо известны Fast-RCNN [3] и Faster-RCNN [4].

Fast-RCNN представляет собой свёрточную искусственную нейронную сеть, включающую в себя как детектор подозрительных зон, так и классификатор найденных объектов. При этом детектирующая сеть является частью классифицирующей сети, поэтому нет необходимости многократного запуска нейронной сети на одном изображении, достаточно лишь одного запуска. Подозрительные зоны находятся на основе откликов нейронной сети на глубоких слоях с помощью процедуры selective search. Весь алгоритм обучается совместно как единая сеть. Данный алгоритм работает существенно быстрее, чем наивные способы детектирования объектов с помощью исчерпывающего поиска, однако время работы остается неприемлемо большим.

Faster-RCNN представляет собой усовершенствование алгоритма Fast-RCNN, призванное решить проблему определения подозрительных зон. Данная архитектура содержит специальный модуль, отвечающий за обнаружение подозрительных зон, который называется Region Proposal Network (RPN). Этот модуль получает на вход карту признаков и для каждой позиции определяет координаты нескольких прямоугольников фиксированного размера, которые используются в качестве подозрительных зон в детекторе. Такие прямоугольники называют якорями (anchors). Детектор представляет собой сеть Fast-RCNN. При этом RPN и Fast-RCNN имеют общие свёрточные слои, что позволяет ускорить обработку изображений.

Для обучения Faster-RCNN используется специальная процедура, состоящая из 4 этапов:

1. Обучается RPN для обнаружения подозрительных зон.
2. Обучается Fast-RCNN с использованием подозрительных зон, которые выдает RPN.
3. RPN переинициализируется весами, полученными при обучении Fast-RCNN. При этом, общая часть двух сетей замораживается.

4. С фиксированными общими слоями окончательно дообучается Fast-RCNN.

Причина введения такой сложной процедуры обучения заключается в том, что стандартная процедура, выделяющая область для классификатора, не является дифференцируемой по декартовым координатам, из-за чего при совместном обучении веса RPN при настройке не будут учитывать направление сдвига координат прямоугольника подозрительной области.

Данные модификации позволили увеличить точность детектирования объектов, а также достичь скорости обработки изображений, равной пяти кадрам в секунду (с применением GPU).

Одноэтапные нейросетевые алгоритмы детектирования объектов на изображении не включают в себя стадию поиска подозрительных зон на изображении, а сразу нацелены на обнаружение объектов. Преимуществом данного вида алгоритмов является их простота и относительно высокая скорость работы, а среди их недостатков можно отметить более низкую точность детектирования объектов по сравнению с двухэтапными алгоритмами, а также меньшую гибкость алгоритма.

Самыми известными представителями данного вида алгоритмов являются архитектуры Yolo [5] и Single Shot detector (SSD) [6].

Детектор Yolo является одним из первых одноэтапных детекторов объектов на изображении, при этом он постоянно модернизировался, поэтому существует три версии этого алгоритма: v1, v2 и v3. На входном изображении наносилась сетка с фиксированным заранее размером ячейки и выбиралось максимальное число B объектов, которые могут присутствовать внутри ячейки. Искусственная нейронная сеть предсказывает для каждой ячейки $4B$ чисел – координаты прямоугольника объекта, центр которого лежит в этой ячейке, значение, определяющее вероятность того, что в данной ячейке присутствует объект, а также по C чисел для каждого объекта – вероятности того, что в ячейке присутствует объект i -го класса при условии, что какой-либо объект в ячейке присутствует (здесь C – количество классов, которые может распознавать детектор).

Дальнейшие версии Yolo содержат модификации, призванные повысить точность автоматического обнаружения объектов и стабильность обучения искусственной нейронной сети. В частности, начиная со второй версии для определения размеров прямоугольников используются заранее созданные шаблонные прямоугольники – якоря. Кроме того, размер входных изображений увеличился и произошли изменения в архитектуре сети, такие как отказ от полносвязных слоев в пользу полносверточной искусственной нейронной сети и добавление слоев батч-нормализации [7]. Также были приняты меры для повышения устойчивости алгоритма к изменениям масштаба объектов.

Детектор SSD основан на идее детектора MultiBox [8], которая заключается в предсказании ограничивающих прямоугольников в разных масштабах. Якоря для SSD выбираются вручную, что позволяет избежать этапа предобучения якорей и позволяет алгоритму работать при любых входных данных. Многомасштабные карты признаков позволяют существенно увеличить точность обнаружения и распознавания объектов, что является большим преимуществом по сравнению с первыми версиями Yolo.

В табл. 1 представлено сравнение современных нейросетевых архитектур полнокадрового обнаружения и распознавания.

Таблица 1

Сравнение современных архитектур полнокадрового распознавания объектов

Ранг	Метод	Скорость (кадров в сек.)	Средняя точность (%)	Название публикации	Год
1	2	3	4	5	6
1	CenterNet ResNet-18	142	28.1	Objects as Points	2019
2	TTFNet	54.4	35.1	Training-Time-Friendly Network for Real-Time Object Detection	2019
3	CenterNet DLA-34	52	37.4	Objects as Points	2019
4	YOLOv3-320	45	28.2	YOLOv3: An Incremental Improvement	2018
5	YOLOv3-418	34	31.0	YOLOv3: An Incremental Improvement	2018
6	CornerNet- Squeeze	33	34.4	CornerNet-Lite: Efficient Keypoint Based Object Detection	2019
7	SSD512- HardNet85	32	35.1	HardNet: A Low Memory Traffic Network	2019
8	RefineDet320 + VoVNet-57	21.2	33.9	An Energy and GPU- Computation Efficient Backbone Network for Real-Time Object Detection	2019
9	YOLOv3-608	20	33	YOLOv3: An Incremental Improvement	2018
10	CenterNet Hourglass-104	14	40.3	Objects as Points	2019
11	NAS-FPNLite MobileNetV2 (7 @ 64)	н/д	25.7	NAS-FPN: Learning Scalable Feature Pyramid Architecture for Object Detection	2019

Применяемый подход. В рамках данной работы для детектирования объектов на изображении использовалась искусственная нейронная сеть архитектуры CenterNet [9]. Данная нейронная сеть относится к классу одноэтапных алгоритмов детектирования объектов. В отличие от многих других детекторов, CenterNet не использует наборы якорей, а обучение признаку размера объекта происходит с помощью регрессии. Положение объектов определяется с помощью предсказания карты ключевых точек, каждая ключевая точка представляет собой центр некоторого объекта. В отличие от других алгоритмов обнаружения объектов, CenterNet не позволяет предсказывать более одного объекта на ячейку карты признаков, поэтому при неудачном выборе размеров выходной карты признаков, некоторые объекты не смогут быть распознанными.

Преимуществом данного алгоритма являются высокая скорость работы, даже по сравнению с другими одноэтапными детекторами, которая, в частности, достигается благодаря тому, что CenterNet позволяет избежать довольно затратной и сложно распараллеливаемой операции постобработки результата Non-Maxima Suppression (NMS), заменяя её еще более простой и быстрой операцией взятия локальных максимумов карты ключевых точек. В связи с вышеперечисленными доводами, было принято решение использовать именно этот нейросетевой алгоритм обнаружения объектов для детектирования надводных объектов в реальном времени.

Подготовка обучающей выборки. Одним из самых важных этапов работы с задачами машинного обучения является подготовка данных для обучения. От качества этих данных очень сильно зависит качество будущей модели, наличие выбросов и некачественная разметка способны подорвать качество автоматического обнаружения настолько, что даже выбор самой лучшей модели не позволит компенсировать низкое качество данных.

В настоящее время существует множество открытых баз данных изображений, содержащих данные для различных задач. В частности, для задачи детектирования судов и других надводных объектов существуют несколько наборов данных, такие как Seaships dataset и singapore maritime dataset.

Seaship Dataset [10] представляет собой набор из 31455 изображений, содержащих шесть основных типов морских судов (перевозчик руды, насыпной грузовой, грузовой корабль, контейнеровоз, рыболовное судно и пассажирское судно). Изображения представляют собой нарезку некоторого числа видеозаписей, снятых с камер, расположенных на берегу реки. Запись происходила в разное время суток с разных мест, однако все корабли расположены примерно с одного ракурса. Однако в открытом доступе находится лишь небольшая часть этого набора данных, которая включает в себя семь тысяч изображений.

База Singapore maritime dataset [11] содержит около 80 видеозаписей морской обстановки, часть из которых снята с палубы судна, остальные сняты с берега. Записи сняты в разное время суток и в разных погодных условиях, однако длительность каждой видеозаписи невелика и составляет порядка одной минуты. Все видеозаписи имеют высокое разрешение (1920x1080). Часть видеозаписей была сформирована в ближнем инфракрасном диапазоне, что может быть полезно для распознавания морских объектов в сложных погодных условиях. Однако на большинстве фрагментов видеозаписей присутствуют лишь минимальные изменения между соседними кадрами, а также для некоторых видеозаписей отсутствует разметка.

В связи с тем, что ни один открытый набор данных не удовлетворяет требованиям, предъявляемым в данной работе, было принято решение подготовить свой набор данных, взяв за основу вышеперечисленные открытые наборы данных, добавив к ним еще около 100 видеозаписей морской обстановки, большая часть из

которых была получена из открытых источников, остальные были сняты вручную. Половина из них представляет записи, снятые с берега, другая половина – с борта крупного морского судна или катера.

После нарезки данных видеозаписей получилось более 10000 изображений, после объединения с данными из других наборов суммарно получилось около 40000 изображений.

Следующим шагом подготовки данных стала разметка еще неразмеченных данных и унификация разметки, т. е. приведение разметок всех частей набора к одному формату и одному списку классов.

Было принято решение распознавать пять классов надводных объектов: малое судно, буй, крупногабаритное судно, военное судно, льдины, а также был введен отдельный класс для сложно идентифицируемых надводных объектов (прочие помехи).

Для осуществления разметки подготовленных данных (т.е. выделения объектов на изображениях и маркирования их меткой класса) было принято решение воспользоваться одним из сервисов автоматизации удалённого выполнения заданий. На данный момент существует несколько подобных сервисов, например Amazon Mechanical Turk [12] или его отечественный аналог – Яндекс. Толока [13]. Выбор был сделан в пользу отечественного сервиса. Данный сервис позволяет удалённо выдавать задания пользователям для выполнения за определенную плату. Имеется набор шаблонов для заданий и возможность реализовать интерфейс задания с помощью языка программирования javascript.

Шаблонное задание для выделения объектов на изображении имеет минимальный функционал и оказалось не очень удобным в рамках данной задачи. Поэтому было принято решение добавить дополнительные возможности, а именно:

1. Окно выбора класса объекта.
2. Дополнительные горячие клавиши вдобавок к базовым.
3. Возможность перемещать прямоугольники с помощью мыши и клавиш-стрелок на клавиатуре.
4. Возможность переносить разметку с одного изображения на следующие.

Так как исходные данные представляли собой видеопоследовательность, то часто соседние изображения отличаются незначительно. Поэтому можно разметить вручную только одно изображение, перенести разметку на остальные, и при необходимости откорректировать её.

Запуск задания происходил в два этапа: на первом этапе задание запускалось в тестовом режиме. На этом этапе происходила отладка функционала, а также приведение всех размеченных наборов данных к требуемой классификации. В течение месяца задания выполняла контрольная группа из 5 человек. В ходе тестирования пользователи из контрольной группы разметили 31550 изображений в течение месяца.

Полученная база данных образов надводных объектов состоит из 42220 изображений, содержащих около 254000 объектов, из них 36440 малых судов, 6822 буюв, 175608 больших судов, 4819 военных судов.

Обучение искусственной нейронной сети. Для проведения экспериментов, собранный набор данных был разделен на три части, одна из которых использовалась для обучения нейронной сети, вторая – для валидации и настройки параметров, а третья использовалась при финальном тестировании. Разделение было проведено таким образом, чтобы каждая часть разбиения содержала изображения из всех наборов данных, входящих в состав данного набора.

Для обучения искусственной нейронной сети была выбрана библиотека PyTorch [14]. Первоначально была реализована функция подготовки изображений, которая включает в себя считывание изображений, аугментацию (добавление в

выборку искусственно сгенерированных изображений с наложением искажений и шумов) и вычисление эталонных карт центров и размеров объектов. В качестве видов аугментации использовались зеркальное отражение, вырезание случайной области изображения, содержащей некоторый объект, зашумление гауссовским шумом, изменения контраста и яркости.

Карты центров объектов представляют собой изображения, в которых единичную яркость имеют только пиксели с координатами, соответствующими центру некоторого объекта данного класса. Количество таких карт для изображения равно количеству классов для распознавания. Для более стабильного обучения применяется неравномерное гауссово размытие карт признаков с помощью прямоугольного ядра, зависящего от размеров объекта. В отличие от подхода из оригинальной статьи, который заключался в размытии с квадратным окном, данное представление карты центров объектов лучше работает для объектов с большим соотношением сторон. Подобные объекты широко представлены в обучающей выборке.

Функция потерь для данной нейронной сети состоит из нескольких слагаемых. Первое слагаемое отвечает за качество детектирования центров объектов. Оно представляет собой модифицированный focal loss [15] и имеет вид:

$$L_k = \frac{-1}{N} \begin{cases} (1 - \widehat{Y}_{xyc})^\alpha \log(\widehat{Y}_{xyc}) & \text{if } Y_{xyc} = 1 \\ (1 - \widehat{Y}_{xyc})^\beta (\widehat{Y}_{xyc})^\alpha & \\ \log(1 - \widehat{Y}_{xyc}) & \text{otherwise} \end{cases}$$

где Y_{xyc} – Истинная карта центров объектов, \widehat{Y}_{xyc} – карта предсказанных центров объектов. Второе слагаемое отвечает за качество определения центров объектов и представляет собой L1 loss, т.е. средний модуль разницы между истинными размерами объектов и предсказанными. Так как большая часть истинной маски размеров объектов заполнена нулями, функция потерь считается не по всей маске, а только по тем позициям, которые являются центром некоторого объекта. Это позволяет избежать деградации сети и предсказания нулевых размеров для всех объектов.

Для повышения скорости работы сети размер карт признаков в 4 раза меньше, чем размер входного изображения, что с одной стороны значительно уменьшает объем вычислений, а с другой стороны не влияет на качество обнаружения объектов, поскольку маловероятно наличие объектов, расстояние между центрами которых меньше четырех пикселей. В оригинальной статье также сеть обучалась предсказывать разницу между истинным центром объекта и положением пикселя, который отвечает за её центр в карте признаков. Однако в данной работе было принято решение от этого отказаться, поскольку использование данной величины лишь незначительно повышает точность распознавания.

В ходе экспериментов были обучены две искусственные свёрточные нейронные сети. Они отличались в основном своими кодерами – слоями, извлекающими признаки с изображений. В одной сети в качестве кодера использовалась архитектура resnet18 [16], архитектура, основанная на residual связях, которые позволяют сети иметь очень большую глубину. Во второй сети использовалась MobileNetv2 [17] – легковесная архитектура, основанная на операции сепарабельной свертки, позволяющей ускорить вычисления и уменьшить количество параметров сети.

В качестве декодера, т.е. той части сети, которая восстанавливает карты центров и размеров объектов по признакам, в обоих вариантах сетей использовалась последовательность из нескольких деформируемых свёрточных слоев с билинейной интерполяцией для повышения пространственных размеров объектов. Деформируемые свёрточные слои – это слои, которые имеют возможность выучить форму своего рецептивного поля, в отличие от обычных свёрточных слоев, в которых она всегда имеет фиксированный размер и форму (квадрат).

Обучение обоих вариантов искусственной нейронной сети производилось в течение 100 эпох (полных циклов обучения по всем данным). Сеть с resnet18 была инициализирована весами сети такой же архитектуры, обученной на открытом наборе данных Coco [18]. В течение первой эпохи в ней обучались только слои, непосредственно восстанавливающие карты центров и размеров объектов, остальные слои были заморожены.

У сети с mobilenet v2 кодер инициализировался весами модели, предобученной на наборе данных ImageNet [19], остальные веса инициализировались случайным образом. В течение первой эпохи веса кодера были заморожены.

В качестве оптимизатора использовался алгоритм Adam [20], позволяющий автоматически корректировать скорость обучения (learning rate) в процессе обучения с помощью накопления информации о градиентах в предыдущие итерации. Базовый learning rate был выбран равным 0.0003, и он уменьшался в 4 раза каждые 20 эпох.

График функции потерь (ошибки) на валидационной выборке представлен на рис. 1.

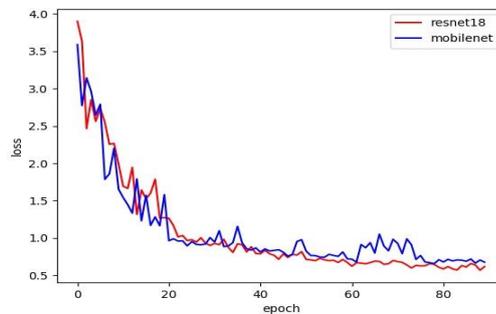


Рис 1. График функции потерь для двух вариантов искусственных сверточных нейронных сетей

В качестве метрик оценки качества обнаружения объектов использовались Mean Average Precision (mAP) и Mean average Recall (mAR).

Mean average precision представляет собой усреднение метрики average precision (AP) для всех порогов IoU от 0.5 до 1 с заданным шагом:

$$mAP = \frac{1}{|iou|} \sum_{thr \in iou} AP(thr).$$

AP для фиксированного порога thr вычисляется следующим образом: для каждого найденного объекта происходит поиск сопоставления среди истинных объектов. Соответствие объявляется истинным, если отношение площади пересечения к площади объединения (Intersection over Union, IoU) предсказанного и истинного объектов больше порога thr. Соответствия сортируются согласно убыванию уверенности сети в предсказанном объекте. Для любого набора из первых k соответствий можно посчитать точность (precision) и полноту (recall):

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

где TP – число верно определенных объектов, FP число ошибочно найденных объектов, FN – число объектов, которые не были найдены. С ростом k полнота монотонно увеличивается, а точность может как увеличиваться, так и уменьшаться. AP определяется как площадь под графиком, отражающим зависимость точности от полноты, и имеет вид:

$$AP = \int_0^1 precision(r)dr.$$

Mean Average Recall представляет собой среднее значение полноты при различных выборах порога для IoU, учитывая только те пороги, которые больше, чем 0.5

$$AP@[IoU = a:b] = 2 \int_a^b recall(IoU)dIoU.$$

Результаты тестирования полученных искусственных свёрточных нейронных сетей на тестовой выборке представлены в табл. 2.

Таблица 2

Результаты тестирования полученных ИСНС

Оцениваемый параметр	ResNet18	MobileNet v2
AP@[IoU=0.5:0.95]	0.226	0.178
AP@[IoU=0.5]	0.645	0.494
AP@[IoU=0.5:0.95] малые объекты	0.169	0.098
AP@[IoU=0.5:0.95] большие объекты	0.304	0.276
AR@[IoU=0.5:0.95]	0.333	0.282
AR@[IoU=0.5:0.95] малые объекты	0.223	0.141
AR@[IoU=0.5:0.95] большие объекты	0.445	0.426
Скорость обработки (GeForce 1060 6GB)	36 к/сек	40 к/сек

Как видно, сеть с кодером ResNet18 работает ощутимо лучше, чем с MobileNet v2, при том, что скорость работы сетей примерно одинаковая. Также стоит отметить, что качество обнаружения малых объектов ниже, чем больших. На рис. 2 представлена генерируемая ИСНС карта предсказаний центров объектов в кадре, а на рис. 3 – результат обработки получаемых карт с выделением распознанных объектов.



Рис. 2. Генерируемая ИСНС карта предсказаний центров объектов в кадре

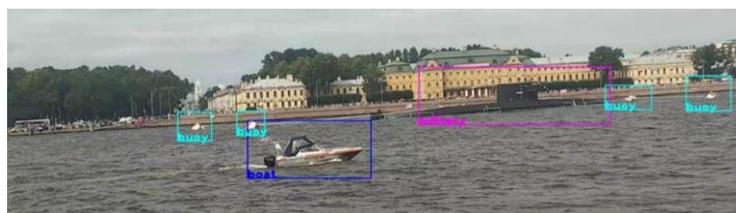


Рис. 3. Результат распознавания объектов в кадре (голубой прямоугольник – буи, синий – малые суда, фиолетовый – военные суда)

Заключение. Резюмируя изложенное, можно сделать следующие выводы.

Представленные алгоритмы обнаружения объектов на изображениях с помощью искусственных свёрточных нейронных сетей доказали свою применимость в решении задач автоматического обнаружения и распознавания надводных объектов в реальном времени и их можно применять при разработке систем технического зрения перспективных систем автоматизации и повышения безопасности гражданского судоходства.

Скорость работы полученных искусственных нейронных сетей достаточна для того, чтобы обрабатывать более 30 кадров в секунду, при условии наличия вычислителя с возможностями массового параллелизма (например GPU или ПЛИС).

Достигнутые точность и полнота распознавания достаточно велики, но не являются предельными, так как полученная база размеченных образов обладает недостаточным качеством разметки ввиду того, что её проводили необученные пользователи. Повысив качество разметки, можно увеличить точность и значительно повысить эффективность автоматического обнаружения и распознавания.

В качестве дальнейших шагов по улучшению алгоритма (помимо повышения объема базы данных образов и качества её разметки) планируется реализовать возможность анализа нескольких предыдущих кадров видеопоследовательности, что позволит улучшить качество распознавания мелких (менее 32x32 пикселей) объектов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Goodfellow I., Bengio Y., Courville A. Deep Learning. MIT Press, 2016, 781 p.
2. Deng L. and Deep D.Yu. Learning: Methods and Applications, *Foundations and Trends in Signal Processing*, 2013, Vol. 7, No. 3–4, pp. 197-387.
3. Girshick R. Fast r-cnn, *Proceedings of the IEEE international conference on computer vision*, 2015, ppP. 1440-1448.
4. Ren S. et al. Faster r-cnn: Towards real-time object detection with region proposal networks, *Advances in neural information processing systems*, 2015, pp. 91-99.
5. Redmon J. et al. You only look once: Unified, real-time object detection, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779-788.
6. Liu W. et al. Ssd: Single shot multibox detector, *European conference on computer vision*, Springer, Cham, 2016, pp. 21-37.
7. Ioffe S., Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift, *arXiv preprint arXiv:1502.03167*, 2015.
8. Szegedy C. et al. Scalable, high-quality object detection, *ArXiv abs/1412.1441*, 2015.
9. Zhou X., Wang D., Krähenbühl P. Objects as points, *ArXiv abs/1904.07850*, 2019.
10. Shao Z. et al. Seaships: A large-scale precisely annotated dataset for ship detection, *IEEE Transactions on Multimedia*, 2018, Vol. 20, No. 10, pp. 2593-2604.
11. Prasad D.K. et al. Video processing from electro-optical sensors for object detection and tracking in a maritime environment: a survey, *IEEE Transactions on Intelligent Transportation Systems*, 2017, Vol. 18, No. 8, pp. 1993-2016.
12. Crowston K. Amazon Mechanical Turk: A Research Tool for Organizations and Information Systems Scholars. In: Bhattacharjee A., Fitzgerald B. (eds) *Shaping the Future of ICT Research. Methods and Approaches*. IFIP Advances in Information and Communication Technology. Vol. 389. Springer, Berlin, Heidelberg, 2012.
13. Yandex Toloka url: <https://toloka.yandex.ru> (accessed 18 February 2020).
14. Paszke A., Gross S., Massa F. and others. PyTorch: An Imperative Style, High Performance Deep Learning Library, *Advances in Neural Information Processing Systems 32*, Curran Associates, Inc., 2019, pp. 8024-8035.
15. Lin T.Y. et al. Focal loss for dense object detection, *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980-2988.

16. He K. et al. Deep residual learning for image recognition, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770-778.
17. Sandler M. et al. Mobilenetv2: Inverted residuals and linear bottlenecks, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, ppP. 4510-4520.
18. Lin T., Maire M., Belongie S.J., Bourdev L.D., Girshick R.B., Hays J., Perona P., Ramanan D., Dollár P. and Zitnick C.L. Microsoft COCO: Common Objects in Context, 2014. DOI: 10.1007/978-3-319-10602-1_48.
19. Deng J., Dong W., Socher R., Li L., Kai Li and Li Fei-Fei. ImageNet: A large-scale hierarchical image database, *2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, 2009*, pp. 248-255.
20. Kingma D.P., Ba J. Adam: A method for stochastic optimization, *arXiv preprint arXiv:1412.6980*, 2014.

Статью рекомендовал к опубликованию к.т.н. П.К. Шульженко.

Тупиков Владимир Алексеевич – АО «Научно-производственное предприятие «Авиационная и морская электроника»; e-mail: tupikov@nppame.ru; 198097, г. Санкт-Петербург, ул. Маршала Говорова, 29, лит. «О», а/я 51; тел.: 88123274667; д.т.н.; профессор; зам. ген. директора; директор научно-производственного комплекса робототехнических систем специального назначения (НПК РТС СН).

Павлова Валерия Анатольевна – e-mail: pavlova@nppame.ru; к.т.н.; зам. директора НПК РТС СН по НИОКР.

Бондаренко Владимир Александрович – e-mail: bondarenko@nppame.ru; начальник центра средств интеллектуальной обработки изображений.

Холод Николай Григорьевич – e-mail: nikkholod@mail.ru; научный сотрудник.

Tupikov Vladimir Alekseevich – SPE "Research and Production Enterprise "Air and Marine Electronics"; e-mail: tupikov@nppame.ru; 198097, Saint Petersburg, street Marshal Govorov, 29, lit. "O", P.O.B 51; phone: 88123274667; dr. of eng. sc.; professor; Deputy Director General; Director of Research and production complex of special purpose robotic systems.

Pavlova Valeria Anatolyevna – e-mail: pavlova@nppame.ru; cand. of eng. sc.; Deputy Director of research at Research and production complex of special purpose robotic systems.

Bondarenko Vladimir Alexandrovich – e-mail: bondarenko@nppame.ru; chief of intellectual image processing center.

Holod Nicolai Grigorievich – e-mail: nikkholod@mail.ru; researcher.

УДК 004.896

DOI 10.18522/2311-3103-2020-1-199-218

В.П. Андреев, В.Л. Ким, С.Р. Эприков

АППАРАТНО-ПРОГРАММНЫЙ ФРЕЙМВОРК ДЛЯ РАЗРАБОТКИ МОДУЛЬНЫХ МОБИЛЬНЫХ РОБОТОВ С ИЕРАРХИЧЕСКОЙ АРХИТЕКТУРОЙ*

В настоящей работе рассматриваются основные проблемы, связанные с необходимостью интеграции разнородных робототехнических компонентов в единую систему при одновременном росте сложности навигационных алгоритмов мобильных роботов. Цель данной работы – представить иерархическую модульную архитектуру для реконфигурируемых мобильных роботов в качестве решения поставленных проблем. В этой архитектуре мобильный робот рассматривается как комбинация модулей, которые в свою очередь состоят из более простых блоков – субмодулей. Каждый субмодуль включает в себя мало-

* Работа выполнена при финансовой поддержке РФФИ (грант №19-07-00892а).