

DOI: <https://doi.org/10.30898/1684-1719.2021.11.14>

УДК: 621.396.96

СРАВНИТЕЛЬНЫЕ РЕЗУЛЬТАТЫ ТОЧНОСТИ КЛАССИФИКАЦИИ РАДИОЛОКАЦИОННЫХ ИЗОБРАЖЕНИЙ ОБЪЕКТОВ НАБОРА MSTAR СВЕРТОЧНЫМИ НЕЙРОННЫМИ СЕТЯМИ С РАЗЛИЧНЫМИ АРХИТЕКТУРАМИ

И. Ф. Купряшкин

Военный учебно-научный центр Военно-воздушных сил «Военно-воздушная академия имени профессора Н.Е. Жуковского и Ю.А. Гагарина»
394064, Воронеж, ул. Старых Большевиков, д. 64

Статья поступила в редакцию 20 октября 2021 г.

Аннотация. Приведены результаты классификации отметок объектов десяти классов (из открытой части набора MSTAR) на радиолокационных изображениях с использованием глубокой сверточной нейронной сети VGG-типа с восемью сверточными слоями. Максимальная точность, достигнутая сетью, составила 97,91%. Кроме того, приведены результаты работы сетей MobileNetV1, Xception, InceptionV3, ResNet50, InceptionResNetV2, DenseNet121, подготовленных с использованием техники переноса обучения. Показано, что в рассматриваемой задаче применение перечисленных предобученных сверточных сетей не дало улучшения точности классификации, составившей от 93,79% до 97,36%. Установлено, что даже ненаблюдаемые визуально локальные особенности фона местности вблизи каждого типа объекта способны обеспечить точность классификации порядка 51% (а не ожидаемые 10% при десятиальтернативной классификации) даже при отсутствии отметок объектов и их теней. Описан порядок подготовки обучающих данных, обеспечивающий исключение влияния фона местности на результат нейросетевой классификации

Ключевые слова: глубокая сверточная нейронная сеть, перенос обучения, радиолокационное изображение, точность классификации.

Abstract. The results of MSTAR objects ten-classes classification using a VGG-type deep convolutional neural network with eight convolutional layers are presented. The

maximum accuracy achieved by the network was 97.91%. In addition, the results of the MobileNetV1, Xception, InceptionV3, ResNet50, InceptionResNetV2, DenseNet121 networks, prepared using the transfer learning technique, are presented. It is shown that in the problem under consideration, the use of the listed pretrained convolutional networks did not improve the classification accuracy, which ranged from 93.79% to 97.36%. It has been established that even visually unobservable local features of the terrain background near each type of object are capable of providing a classification accuracy of about 51% (and not the expected 10% for a ten-alternative classification) even in the absence of object and their shadows. The procedure for preparing training data is described, which ensures the elimination of the influence of the terrain background on the result of neural network classification.

Key words: deep convolutional neural network, transfer learning, radar image, classification accuracy.

Введение.

В настоящее время глубокие нейронные сети стали одним из наиболее широко применяемых инструментов решения ряда задач, традиционно рассматриваемых как задачи систем искусственного интеллекта: распознавание и синтез речи и изображений, обработка текстов, в том числе машинный перевод, компьютерного зрения и т.п. [1-3].

Вряд ли на сегодняшний день существуют области исследований, так или иначе связанные с распознаванием образов, применение глубоких нейросетей в которых не продемонстрировало высокую эффективность, превышающую или, по меньшей мере, не уступающую эффективности традиционных подходов.

Не исключением стала и радиолокация. Обширный обзор состояния дел в данной области (более 200 источников) представлен в работе специалистов Германского Центра Авиации и Космонавтики (German Aerospace Center, DLR), имеющих значительный опыт и внесших существенный вклад в создание систем радиолокационного зондирования Земли и развитие алгоритмов обработки данных, получаемых с их помощью [4]. Авторы [4] выделяют шесть основных

направлений применения технологий глубокого обучения в задачах обработки данных радиолокационной съемки:

- классификация участков поверхности, в том числе поляриметрическая;
- поиск объектов на радиолокационных изображениях (РЛИ);
- оценка характеристик поверхности;
- сглаживание спекл-шума;
- интерферометрическая обработка;
- определение соответствия участков поверхности путем сопоставления их радиолокационных и оптических снимков.

Помимо перечисленных направлений, глубокие нейронные сети продемонстрировали способность решать и другие задачи, такие, как перенос стиля для прогнозирования радиолокационных изображений по имеющимся оптическим [5], синтез РЛИ по зарегистрированному эхосигналу поверхности без использования традиционных алгоритмов [6], синтез радиолокационных портретов целей при инверсном синтезировании апертуры [7] и др.

С точки зрения задач радиолокационного мониторинга поверхности наибольший интерес представляет оценка возможностей глубоких нейросетей по поиску и классификации объектов на РЛИ. В связи с этим как пионерские [8,9], так и основная доля публикуемых на сегодняшний день работ [10-19] посвящены изучению возможностей и поиску эффективных архитектур и гиперпараметров глубоких нейросетей для решения именно этой задачи. Как правило, в качестве основного инструмента нейросетевой классификации рассматриваются глубокие сверточные нейронные сети (ГСНС).

Одна из первых архитектур сверточных сетей LeNet5, продемонстрировавшая высокую эффективность в задаче распознавания рукописных символов, описана в [20]. Повышению роста интереса к этим сетям и толчком к их бурному развитию способствовала победа сверточной сети AlexNet [21] в соревновании ImageNet в 2012 г с большим отрывом по точности классификации.

Результаты сравнения точности классификации объектов с помощью алгоритмов, рассматриваемых как традиционные (K-Neighbor, SVM, AdaBoost, Random Forest, CART и др.), и алгоритмов, построенных на основе ГСНС, демонстрируют уверенное преимущество последних даже при отсутствии предобработки изображений [8, 22-24], а также их меньшую чувствительность к случайным смещениям или поворотам изображения. Особенно хотелось бы отметить работу отечественных авторов [24], включающую детальное описание условий сравнительной оценки для широкого набора алгоритмов.

К сожалению, библиография отечественных работ, посвященных обработке данных радиолокационной съемки с помощью ГСНС, является не очень обширной. Известная работа по вопросам применения нейросетевых технологий в радиолокации [25], несмотря на широкий круг рассмотренных вопросов, создана в период, предшествующий бурному развитию технологий глубокого обучения. Помимо уже отмеченной работы [24] можно отметить краткий обзор [26], а также работы [27, 28], посвященные использованию ГСНС при поиске отметок объектов на РЛИ.

В целом, из анализа известных работ следует вывод, что ГСНС находят все более широкое применение в разнообразных задачах автоматической обработки РЛИ, и в первую очередь – в задачах автоматического поиска и классификации отметок объектов в ходе ведения видовой радиолокационной разведки. Возможно, что уже в ближайшей перспективе именно они станут основным инструментом их решения, так как даже сегодня в ряде случаев они демонстрируют превосходство над человеком.

В связи с этим основной целью работы является сравнительная оценка точности классификации объектов военной техники с помощью глубоких сверточных нейронных сетей, построенных на основе современных архитектурных решений.

1. Характеристика набора обучающих данных.

В качестве исходных данных использован набор MSTAR (Moving and Stationary Target Acquisition and Recognition) [29], с 90-х годов являющийся базовым для создания и изучения алгоритмов распознавания отметок объектов на РЛИ. В некотором роде, он является аналогом широко известного набора изображений рукописных цифр MNIST [30], с которого очень часто начинается обучение вопросам машинного обучения в задачах компьютерного зрения.

Используемая в настоящей работе открытая часть набора MSTAR включает РЛИ десяти образцов военной техники, в основном советского производства. Съемка осуществлялась с помощью РЛС X-диапазона, заявленное пространственное разрешение составляет около 0,3 м.

Каждое изображение представлено в форме массива комплексных отсчетов, записанных в отдельные файлы в собственном формате.

Примеры радиолокационных изображений объектов набора MSTAR и их фотографии приведены на рис. 1.

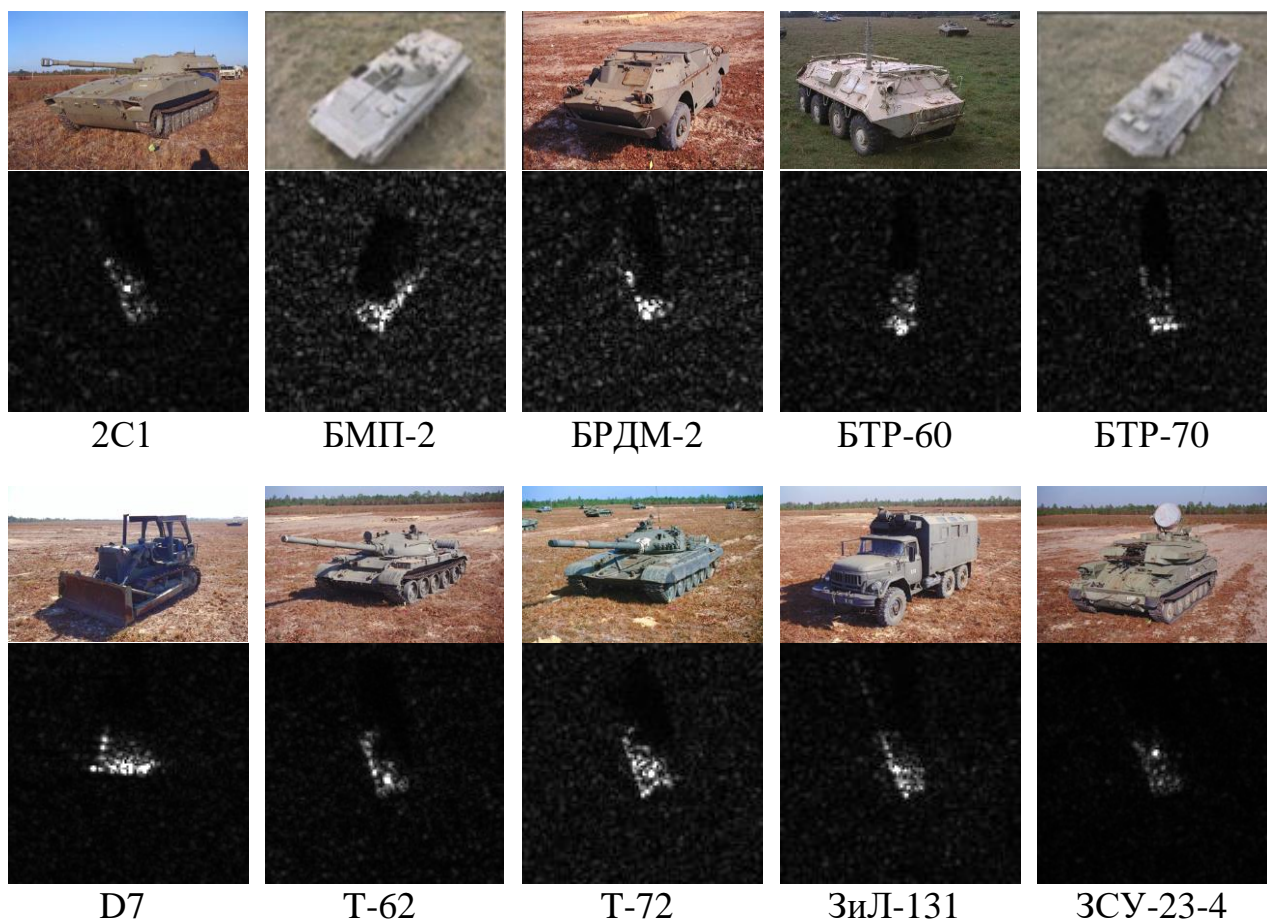


Рис. 1. Оптические и радиолокационные изображения объектов набора MSTAR.

Из всех имеющихся изображений были сформированы обучающий, проверочный и тестовой наборы, характеристики которых приведены в таблице 1. Для всех объектов изображения обучающих и проверочных наборов получены при съемке с углом визирования 17 градусов, изображения тестовых наборов – с углом визирования 15 градусов. Такое распределение изображений по наборам на сегодняшний день является стандартным при исследовании систем распознавания на данных MSTAR, и в иностранной литературе носит обозначение SOC (Standard Operating Conditions) [9].

Общее количество изображений обучающего набора составило 1923, проверочного – 891 и тестового – 2503. С точки зрения обучения ГСНС этот объем является относительно небольшим, кроме того, имеется некоторый дисбаланс классов. Тем не менее, результаты, получаемые на его основе, достаточно информативны для формулирования основных выводов в соответствии с целями работы.

Таблица 1. Состав обучающего, проверочного и тестового наборов.

Объект	Набор	Количество изображений	Объект	Набор	Количество изображений
2С1	Обучающий	209	Т-62	Обучающий	200
	Проверочный	90		Проверочный	99
	Тестовый	274		Тестовый	273
БМП-2	Обучающий	163	Т-72	Обучающий	200
	Проверочный	70		Проверочный	79
	Тестовый	195		Тестовый	274
БРДМ-2	Обучающий	209	Зил-131	Обучающий	200
	Проверочный	89		Проверочный	99
	Тестовый	274		Тестовый	274
БТР-60	Обучающий	179	ЗСУ-23-4	Обучающий	200
	Проверочный	77		Проверочный	99
	Тестовый	195		Тестовый	274
БТР-70	Обучающий	163	D7	Обучающий	200
	Проверочный	70		Проверочный	99
	Тестовый	196		Тестовый	274

2. Подготовка набора обучающих данных, архитектура и результаты работы ГСНС С-CNN.

Как отмечалось выше, каждое изображение набора MSTAR представляет собой массив комплексных отсчетов $\mathbf{\bar{I}}$, причем его размерность для разных классов объектов является различной. Так как минимальная размерность составляет 128×128 , в ходе предварительной подготовки все изображения приводились к этой размерности (обрезались) с последующим вычислением амплитуд комплексных отсчетов, нормировкой и преобразованием в целочисленный 8-битный формат

$$I_{ij} = \begin{cases} \bar{I}_{ij}, & \bar{I}_{ij} \leq 2^8 \\ 2^8, & \bar{I}_{ij} > 2^8 \end{cases}, \quad (1)$$

где $\bar{I}_{ij} = \text{int} \left\{ 2^9 \times \frac{|\dot{I}_{ij}| - I_{\min}}{I_{\max}} \right\}$; $I_{\max} = \max\{|\dot{\mathbf{I}}|\}$ и $I_{\min} = \min\{|\dot{\mathbf{I}}|\}$ – максимальная и

минимальная амплитуды отсчетов соответственно; $i = \overline{1, N_r}$; $j = \overline{1, N_l}$; $N_r = N_l = 128$.

Перед подачей на вход нейронной сети целочисленные значения диапазона $[0; 255]$ преобразовывались в вещественные диапазона $[0; 1]$ путем деления на 255.

Разработка и обучение сети осуществлялось с использованием библиотеки глубокого обучения Keras [30] и фреймворка TensorFlow 2.0 [31].

Архитектура сети включает последовательности четырех пар сверточных слоев (Convolution Layers) с одинаковыми ядрами 3×3 и единичным шагом свертки. После каждой пары включены слои подвыборки (MaxPooling) 2×2 . Первая пара сверточных слоев включает по 32 фильтра, вторая – по 64, третья – по 128 и четвертая – по 256. Полносвязный классификатор включает входной слой на 4096 входов и 256 выходов, слой 50-процентного прореживания (Dropout) и выходной слой с 10 выходами по числу классов. В качестве функции активации сверточных слоев и входного слоя классификатора используется кусочно-линейная (ReLU). Размерность входного изображения составляет

128×128. Для описанной архитектуры количество настраиваемых параметров сети составляет 2'223'082.

При обучении использовался адаптивный алгоритм Adam с параметром скорости обучения 10^{-4} (по умолчанию это значение составляет 10^{-3}), обучение осуществлялось в течение двухсот эпох, размер минипакета составлял 64 изображения. В качестве функции потерь использовалась перекрестная энтропия [31]. Общее время обучения с использованием GPU (GeForce GTX1660SUPER) составило около 25-и минут, т.е. примерно по семь секунд на эпоху.

Веса сети сохранялись после каждой эпохи в отдельных файлах, по окончании обучения осуществлялась их поочередная загрузка и подача тестового набора на вход сети. Набор весов, при котором сеть демонстрировала максимальную точность классификации, принимался в качестве окончательного результата обучения. Несмотря на то, что этот подход, по сути, является неявной подстройкой нейросети уже под тестовый набор, его дальнейшее применение в ходе исследований для всех сетей, рассмотренных ниже, обеспечило получение оценок максимальной достижимой точности классификации в различных условиях.

Архитектура описанной сети, для которой далее в работе принято обозначение C-CNN (Custom CNN), показана на рис. 2.

Представление архитектуры на рис. 2б приведено в нотации библиотеки Keras, широко используемой в публикациях по данной теме. В соответствии с ней каждая строка соответствует слою ГСНС: «conv2d_9 (Conv2D) (None, 126, 126, 32) 320» – сверточному слою с 32-мя картами признаков, размерностью выхода 126×126 и 320 настраиваемыми параметрами (весами); «activation_9 (Activation) (None, 126, 126, 32) 0» – слою activation_9 соответствующих функций активации (нелинейностей) выходов предыдущего сверточного слоя; «max_pooling2d_5 (MaxPooling2 (None, 62, 62, 32) 0» – пулинговому слою (слою подвыборки) с размерностью 2×2, 32 выходными картами признаков размерностью 62×62 и т.д. Слой «flatten_2 (Flatten) (None, 4096) 0» соответствует слою, преобразующему набор 256-и двумерных массивов с размерностью 4×4 в одномерный массив размерности 4096 для подачи на вход полносвязного слоя; слой «dense_3 (Dense) (None, 256) 1048832» – полносвязному слою на 4096

входов и 256 выходов с 1048832 настраиваемыми параметрами, слой «dropout_2 (Dropout) (None, 256) 0 – слою прореживания; слой «dense_4 (Dense) (None, 10) 2570» – полносвязному слою на 256 входов и 10 выходов с 2570 настраиваемыми параметрами. Ход процесса обучения иллюстрируют зависимости точности и потерь на обучающем и проверочных наборах (рис. 3).

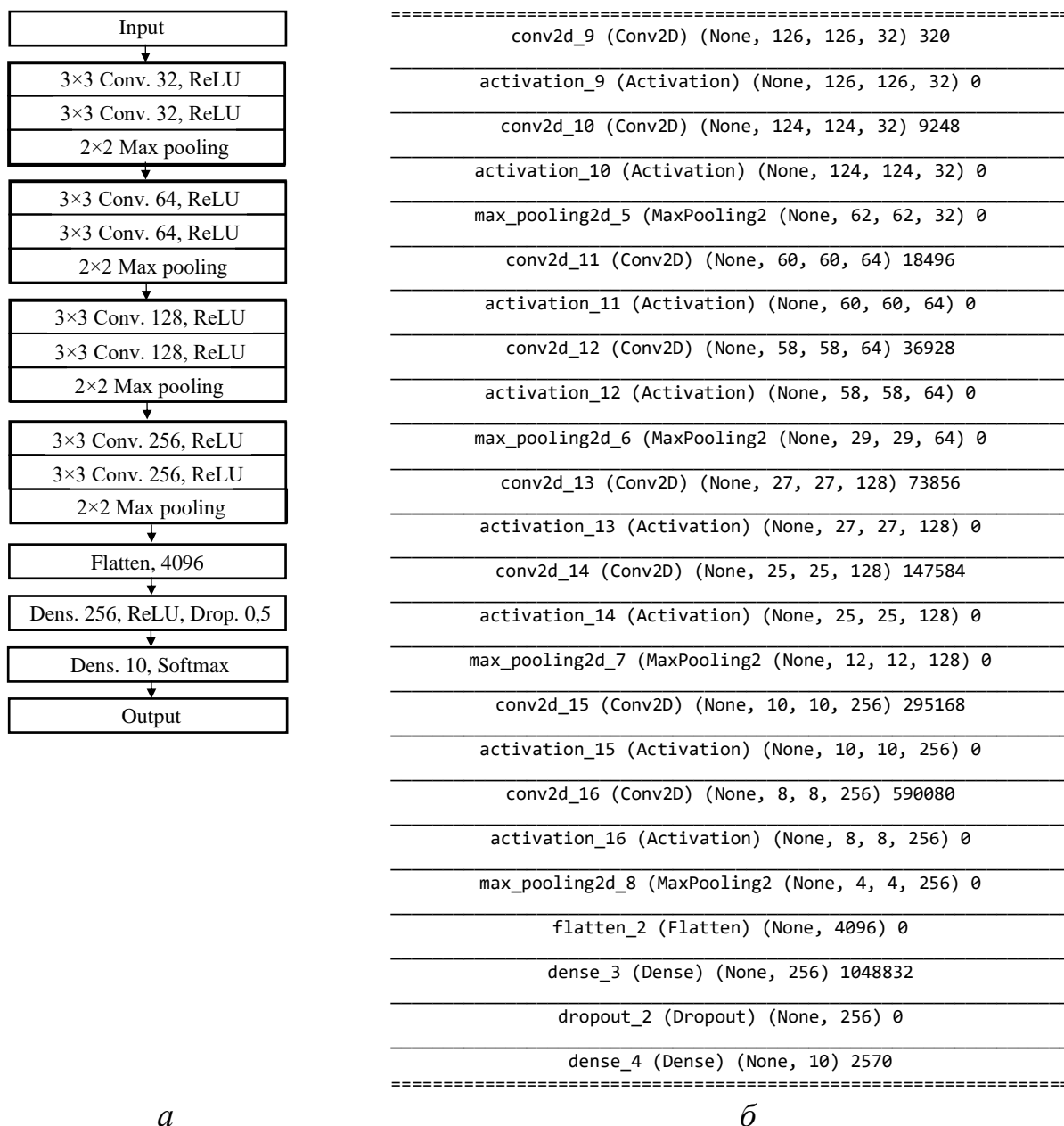


Рис. 2. Архитектура сети C-CNN в форме схемы (а) и в нотации библиотеки Keras (б).

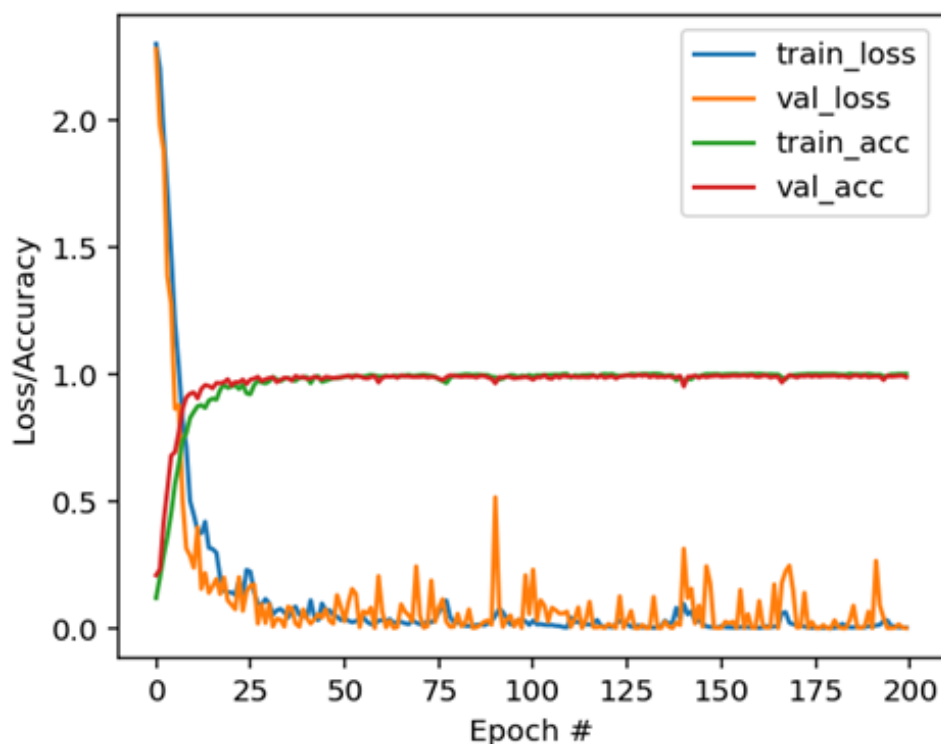


Рис. 3. Характеристика процесса обучения сети C-CNN: train-acc – точность на обучающем наборе, val-acc – точность на проверочном наборе, train-loss – значения функции потерь на обучающем наборе, val-loss – значения функции потерь на проверочном наборе.

Наилучшую точность классификации показала сеть с весами, полученными на 131-й эпохе – 98,12%, то есть из общего объема тестового набора в 2503 изображений неверно классифицированными оказались 47.

Особенностью обучения явилось то, что в ходе подготовки всех наборов не учитывалась неоднородность фона местности и его уникальность в районе расположения объекта каждого класса. Для оценки возможностей сети по решению задачи классификации при отсутствии отметок собственно объектов из имеющегося был сформирован новый набор данных, центральным областям изображений которого (размером 100×100 отсчетов) были присвоены нулевые значения, т.е.

$$\dot{I}_{ij} = \begin{cases} 0, & i \in [N/2 - N_c/2; N/2 + N_c/2] \wedge j \in [N/2 - N_c/2; N/2 + N_c/2] \\ \dot{I}_{ij}, & \text{в остальных случаях} \end{cases}, \quad (2)$$

при $N_c = 100$, после чего они были преобразованы в целочисленный формат в соответствии с (1).

Примеры изображений сформированного набора данных приведены на рис. 4.

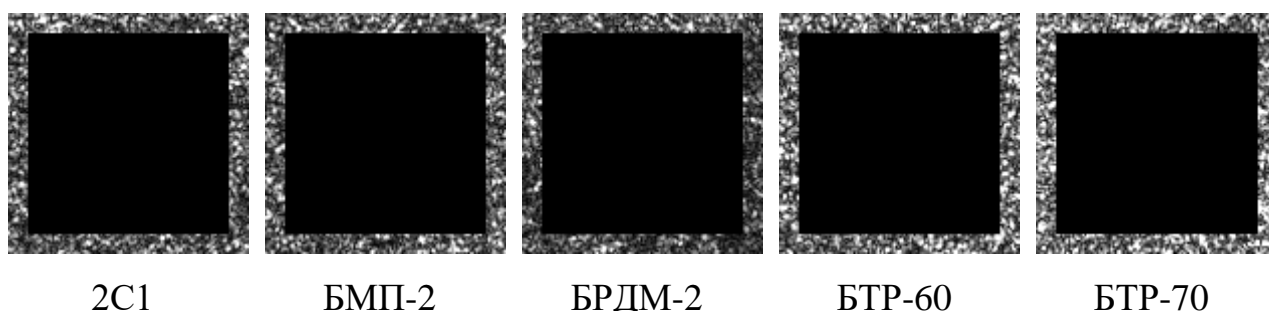


Рис. 4. Примеры изображений с исключенными областями отметок объектов.

Как видно, изображения полученного набора не включают отметки объектов и контуры их радиолокационных теней, а сколько-нибудь заметные отличия статистических характеристик фона на них не прослеживаются.

Тем не менее, по результатам обучения сети С-CNN при тех же условиях была достигнута точность правильной классификации порядка 50,72%, т.е. практически в половине случаев локальные свойства фона местности способствовали его правильной классификации и в тестовом наборе. Иначе, сеть «приучилась» к локальным свойствам участков местности, на которых были расположены объекты классификации. Необходимо отметить, что практически во всех работах, посвященных оценке возможностей различных алгоритмов, и не только нейросетевых, по классификации радиолокационных отметок объектов набора MSTAR, это обстоятельство не учитывается.

Для исключения влияния фона в последующем вычисленные амплитуды отсчетов исходных комплексных изображений сравнивались с порогом

$$I_{ij} = \begin{cases} |I_{ij}|, & |I_{ij}| \geq I_0 \\ 0, & |I_{ij}| < I_0 \end{cases}, \quad (3)$$

где пороговое значение $I_0 = I_{\phi_n}$ определялось как значение n -го элемента упорядоченной по возрастанию векторизированной выборки фоновых отсчетов

$I_{\phi(i-1)N_l+j} = \dot{I}_{ij}$ из области изображения, заведомо не включающей отсчеты отметки объекта и его радиолокационной тени.

В качестве такой области принимались участки изображения с интервалами индексов $i \in [1; N_r]$, $j \in [1; N_\phi]$ и $i \in [1; N_r]$, $j \in [N_l - N_\phi + 1; N_\phi]$ (заштрихованные области на рис. 5). Значение индекса определялось в соответствии с правилом $n = \text{int}\{2pN_rN_\phi\}$, где $p \in [0;1]$ приблизительно соответствует вероятности превышения порога I_0 амплитудой фонового отсчета (ложной тревоге).

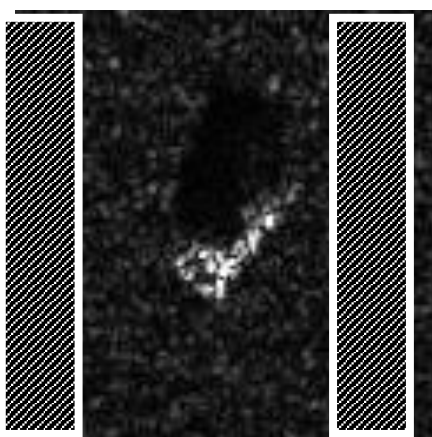


Рис. 5. Области фоновых отсчетов.

Так как при отсутствии фона размерность изображений 128×128 отсчетов с точки зрения классификации объектов является избыточной, в дальнейшем при подготовке наборов данных она понижалась до 44×44 путем осуществления операции вида

$$\dot{I}_{ij} = \dot{I}_{i+\text{int}\{N_r/2-N/2\}, j+\text{int}\{N_l/2-N/2\}}, \quad (4)$$

где $i = \overline{1, N}$ и $j = \overline{1, N}$ при $N = 44$.

Результат пороговой обработки (3) комплексного изображения (4) преобразуется в целочисленный 8-битный формат в соответствии с (1).

Примеры изображений набора, сформированного при $N_\phi = 30$ и $p = 0,01$, приведены на рис. 6.

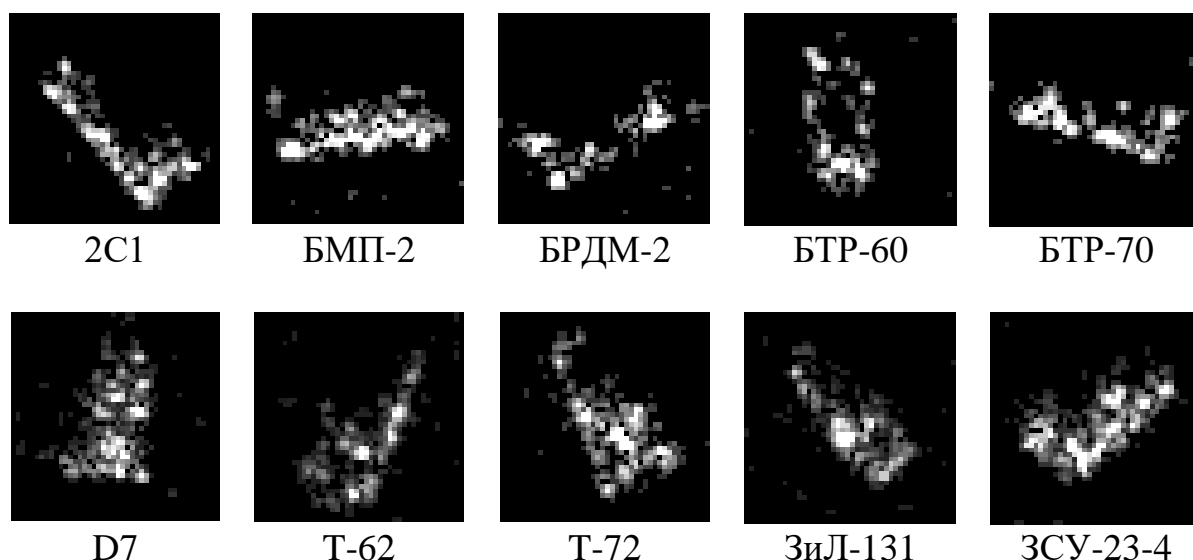


Рис. 6. Примеры изображений набора данных с подавленными фоновыми отсчетами.

Ход обучения иллюстрируют зависимости на рис. 7. В качестве дополнительного приема предотвращения переобучения применялось расширение данных (augmentation) путем смещения каждого изображения по вертикали и горизонтали на случайное число пикселей от одного до пяти и двукратной подачей обучающего набора на вход сети в течение каждой эпохи. Необходимо отметить, что задача расширения наборов данных в случае радиолокационных изображений является весьма актуальной вследствие их гораздо меньшего объема по сравнению с оптическими. В связи с этим на сегодняшний день существует довольно много работ, посвященных этой проблеме [28, 32-35]. Основные подходы состоят в преобразовании имеющихся двумерных портретов объектов для измененных пространственных условий съемки [32] или в моделировании портретов на основе расчета поля рассеяния объектов с использованием их 3D-моделей [28, 34].

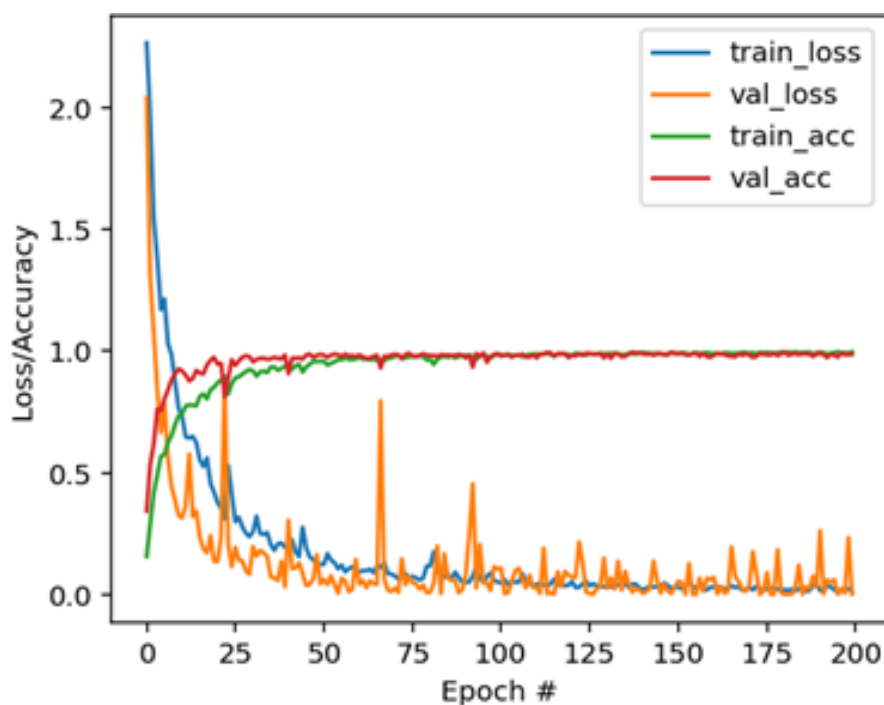


Рис. 7. Характеристика процесса обучения сети C-CNN на наборе данных с подавленными фоновыми отсчетами.

Наилучшую точность классификации 97,91% (52 неверно классифицированных изображения из 2503) показала сеть с весами, полученными на 163-й эпохе.

Из сравнения точностей, достигнутых сетью при обучении на наборах данных с фоном и без, однозначный вывод о выраженном влиянии фона на результат классификации сделать сложно, так как различие не превышает 0,21%. В большей степени такое значение может быть обусловлено относительно малыми объемами обучающего и тестового наборов. Тем не менее, тот факт, что даже при отсутствии отметок объектов и их теней точность классификации изображений достигает 50%, свидетельствует о целесообразности пороговой обработки при формировании наборов данных и в дальнейшем.

При выборе архитектуры сети осуществлялся перебор различных вариантов сочетания гиперпараметров (количества и размерности сверточных слоев, типов функций активации, включение слоев пакетной нормализации (BatchNormalization), слоев прореживания (Dropout) на различных этапах и др.), а также сравнение скорости и качества обучения при различных способах инициализации весов, типах и параметрах оптимизаторов (Adam, RMSProp,

Adadelta). В целом, заметного улучшения точности классификации при возросших временных затратах включение дополнительных этапов обработки не показало.

Так или иначе, для использованных относительно небольшого набора данных и аппаратной платформы описанная архитектура обеспечила достаточно высокую скорость обучения при точности классификации, сопоставимой с достигаемой аналогичными сетями, и в то же время превышающей точности, достигаемые другими (не нейросетевыми) алгоритмами классификации [24]. Необходимо отметить, что результаты классификации с помощью сетей [9, 15] хотя и превышают 99%, т.е. демонстрируют на 1-1,5% более высокую точность по сравнению с C-CNN, предварительная обработка данных не включала подавление фона. Вполне вероятно, что на описанном наборе их точность несколько снизится.

3. Результаты работы ГСНС с известными архитектурами, обученными с использованием техники переноса обучения.

Несмотря на то, что сеть C-CNN представляет собой классическую сверточную сеть VGG-типа [36], достигнутая точность классификации 97,91% является довольно высокой. Тем не менее, на сегодняшний день существует значительное разнообразие более сложных архитектур, включающих bottleneck-слои (сети типа Inception), residual-блоки (сети типа ResNet), а также ряд других решений и их сочетаний.

Как правило, эти сети обучают и тестируют на наборе данных цветных изображений ImageNet, используемом в проекте ImageNet Large Scale Visual Recognition Challenge (ILSVRC), в рамках которого различные программные продукты ежегодно соревнуются в классификации и распознавании объектов и сцен в базе данных ImageNet¹.

¹ По состоянию на август 2021 года первенство в соревновании ILSVRC удерживает сеть [ViT-G/14](#) с примерно двумя миллиардами настраиваемых параметров.

Хотя программные реализации этих сетей находятся в открытом доступе, набор данных MSTAR для их обучения характеризуется недостаточным объемом. В этом случае целесообразно применение техники переноса обучения (Transfer Learning), суть которой состоит в использовании сверточных слоев уже обученной сети и замене ее полносвязного классификатора на собственный с последующим дообучением полученной сети на своем наборе данных [30, 37].

В качестве предобученных сетей выбирались наиболее популярные архитектуры ГСНС, появление каждой из которых являлось знаковым с точки зрения новизны и эффективности использованных решений: InceptionV3, ResNet50, InceptionResNetV2, Xception, DenseNet121, MobileNetV1. Краткие сведения о них приведены в таблице 2.

Таблица 2. Характеристика популярных ГСНС.

Наименование	Год	Количество слоев	Количество настраиваемых параметров	Выходное количество карт	Top-1 Accuracy
InceptionV3	2015	48	27,1М	2048	78,8
ResNet50	2015	50	23,6М	2048	79,16
InceptionResNetV2	2016	164	55,8М	1536	80,1
Xception	2016	36	22,8М	2048	79
DenseNet121	2016	121	7,9М	1024	74,98
MobileNetV1-128	2017	28	4,2М	1024	64,4

При дообучении всех сетей использовался оптимизатор Adam с параметром скорости обучения 10^{-5} , классификатор включал входной полносвязный слой на 512 выходов, слой 50-процентного прореживания и выходной полносвязный слой на 10 выходов по числу классов набора MSTAR.

InceptionV3.

Особенностью сетей типа Inception является использование Inception-блоков [38], т.е. использование в одном слое сверточных фильтров с различной размерностью, включая 1×1 , $1 \times n$, $n \times 1$, с последующим объединением [39].

Размерность входа сети 128×128 отсчетов, входа классификатора – 8192. Общее количество настраиваемых параметров с учетом классификатора – $25'968'298$. Характеристику архитектуры и параметров слоев сети иллюстрирует рис. 8.

```

=====
inception_v3 (Model) (None, 2, 2, 2048) 21802784
-----
flatten_1 (Flatten) (None, 8192) 0
-----
dense_1 (Dense) (None, 512) 4194816
-----
dropout_1 (Dropout) (None, 512) 0
-----
dense_2 (Dense) (None, 10) 5130
=====
    
```

Рис. 8. Архитектура сети на основе InceptionV3 (в нотации библиотеки Keras).

Наилучшую точность обучения 94,83% на тестовом наборе продемонстрировала сеть с набором весов, полученным на 191-й эпохе.

ResNet50.

Особенность семейства сетей ResNet состоит во введении остаточных блоков (residual blocks), содержащих прямое соединение, обходящее один или большее количество слоев [40]. Это позволило существенно уменьшить остроту проблемы исчезающих градиентов и повысить точность работы сети. Другой особенностью сетей ResNet является использование пакетной нормализации (batch-normalization) в каждом слое. Успешность такого решения обусловила появление большого количества модификаций сети (xResNet, SE-ResNet, ResNeXt и др.). Размерность входа сети 64×64 отсчета, входа классификатора – 2048. Общее количество настраиваемых параметров с учетом классификатора – $24'588'810$. Характеристику архитектуры и параметров слоев сети иллюстрирует рис. 9.

Наилучшую точность обучения 94,59% на тестовом наборе продемонстрировала сеть с набором весов, полученным на 165-й эпохе.

```

=====
resnet50 (Model) (None, 2, 2, 2048) 23587712
-----
max_pooling2d_2 (MaxPooling2 (None, 1, 1, 2048) 0
-----
flatten_1 (Flatten) (None, 2048) 0
    
```

dense_1 (Dense) (None, 512)	1049088
dropout_1 (Dropout) (None, 512)	0
dense_2 (Dense) (None, 10)	5130

Рис. 9. Архитектура сети на основе ResNet50 (в нотации библиотеки Keras).

InceptionResNetV2.

Особенностью сети является использование блоков Residual Inception, т.е. блоков Inception с дополнительным прямым соединением [41].

Размерность входа сети 128×128 отсчетов, входа классификатора – 6144. Общее количество настраиваемых параметров с учетом классификатора – 57’427’562. Характеристику архитектуры и параметров слоев сети иллюстрирует рис. 10.

inception_resnet_v2 (Model) (None, 2, 2, 1536)	54336736
flatten_1 (Flatten) (None, 6144)	0
dense_1 (Dense) (None, 512)	3146240
dropout_1 (Dropout) (None, 512)	0
dense_2 (Dense) (None, 10)	5130

Рис. 10. Архитектура сети на основе InceptionResNetV2 (в нотации библиотеки Keras).

Наилучшую точность обучения 93,79% на тестовом наборе продемонстрировала сеть с набором весов, полученным на 121-й эпохе.

Xception.

Особенностью сети является использование так называемых экстремальных Inception-блоков, т.е. блоков Inception с фильтрами размерностью только 3×3 и 1×1 [42].

Размерность входа сети 128×128 отсчетов, входа классификатора – 8192. Общее количество настраиваемых параметров с учетом классификатора – 25’006’898. Характеристику архитектуры и параметров слоев сети иллюстрирует рис. 11.

```

=====
xception (Model) (None, 4, 4, 2048) 20861480
-----
max_pooling2d_1 (MaxPooling2 (None, 2, 2, 2048) 0
-----
flatten_1 (Flatten) (None, 8192) 0
-----
dense_1 (Dense) (None, 512) 4194816
-----
dropout_1 (Dropout) (None, 512) 0
-----
dense_2 (Dense) (None, 10) 5130
=====
    
```

Рис. 11. Архитектура сети на основе Xception (в нотации библиотеки Keras).

Наилучшую точность обучения 97,36% на тестовом наборе продемонстрировала сеть с набором весов, полученным на 178-й эпохе.

DenseNet121.

Особенностью сети является наличие в отдельном блоке не одного дополнительного прямого соединения (как в ResNet50 или InceptionResNetV2), а нескольких прямых соединений, причем выход каждого слоя подается на входы всех последующих слоев [43]. Размерность входа сети 64×64 отсчета, входа классификатора – 1024. Общее количество настраиваемых параметров с учетом классификатора – 7'483'786. Характеристику архитектуры и параметров слоев сети иллюстрирует рис. 12.

```

=====
densenet121 (Model) (None, 2, 2, 1024) 7037504
-----
max_pooling2d_1 (MaxPooling2 (None, 1, 1, 1024) 0
-----
flatten_1 (Flatten) (None, 1024) 0
-----
dense_1 (Dense) (None, 512) 524800
-----
dropout_1 (Dropout) (None, 512) 0
-----
dense_2 (Dense) (None, 10) 5130
=====
    
```

Рис. 12. Архитектура сети на основе DenseNet121 (в нотации библиотеки Keras).

Наилучшую точность обучения 96,47% на тестовом наборе продемонстрировала сеть с набором весов, полученным на 198-й эпохе.

MobileNetV1.

Сеть характеризуется наименьшим количеством параметров и создана с целью использования на мобильных устройствах [44]. Принцип построения сходен с принципом построения сети Xception, и состоит в осуществлении свертки входных карт признаков с фильтром с размерностью 3×3 , с последующей C_{out} -кратной сверткой фильтром 1×1 , где C_{out} – количество выходных карт признаков.

Размерность входа сети 128×128 отсчетов, входа классификатора – 4096. Общее количество настраиваемых параметров с учетом классификатора – $5'309'770$. Характеристику архитектуры и параметров слоев сети иллюстрирует рис. 13.

```

=====
mobilenet_1.00_128 (Model) (None, 4, 4, 1024) 3228864
-----
max_pooling2d_1 (MaxPooling2 (None, 2, 2, 1024) 0
-----
flatten_1 (Flatten) (None, 4096) 0
-----
dense_1 (Dense) (None, 512) 2097664
-----
dropout_1 (Dropout) (None, 512) 0
-----
dense_2 (Dense) (None, 10) 5130
=====

```

Рис. 13. Архитектура сети на основе MobileNetV1 (в нотации библиотеки Keras).

Наилучшую точность обучения 96,47% на тестовом наборе продемонстрировала сеть с набором весов, полученным на 103-й эпохе.

Сравнительная диаграмма точности классификации, достигнутой всеми перечисленными сетями, приведена на рис. 14.

Точность классификации более 97% продемонстрировали сети C-CNN и Xception, более 96% – сети MobileNetV1 и DenseNet121. Так как обучение сетей ResNet50 и DenseNet121 осуществлялось на платформе без GPU, размерность входного изображения для них снижалась со 128×128 до 64×64 . Сети MobileNetV1 и DenseNet121 показали одинаковый результат 96,47%, т.е. одинаковое количество неправильно классифицированных целей – 88. Такое

совпадение не является чем-то исключительным, т.к. общий объем тестового набора в 2503 изображения является относительно небольшим.

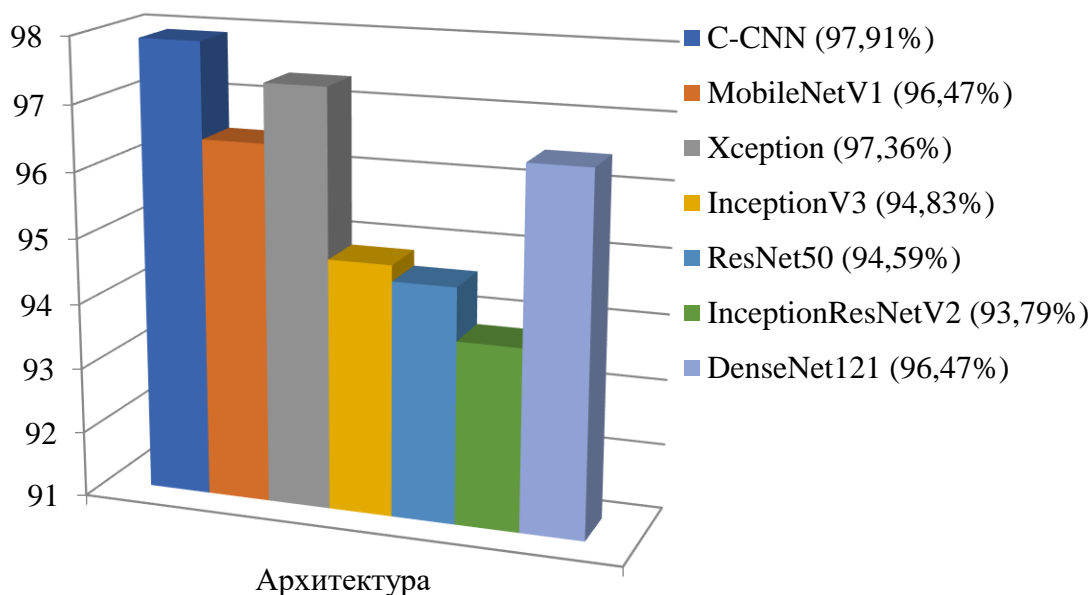


Рис. 14. Сравнительная характеристика точности классификации ГСНС с различной архитектурой.

Заключение.

Применение ГСНС в задаче классификации РЛИ объектов военной техники продемонстрировало высокую точность. При распознавании отметок объектов десяти классов на фоне местности точность составила 98,12%. При этом установлено, что точность классификации соответствующих изображений с удаленными отметками и тенями объектов составила величину не около 10%, как ожидалось, а около 51%, т.е. даже при отсутствии отметок в половине случаев задача классификации решалась верно. Скорее всего, это обусловлено тем, что ГСНС «выучила» практически ненаблюдаемые визуально локальные особенности фона местности, на котором был расположен тот или иной объект. После подавления фона точность классификации снизилась до 97,91%, т.е. на 0,21%. Такое незначительное ухудшение может являться как следствием подавления фона, так и флюктуацией процесса обучения. Тем не менее, отсюда следует вывод об определяющем вкладе в результат классификации вида именно самой отметки, а не ее тени или фона. И хотя в подавляющем большинстве

известных публикаций, посвященных вопросам распознавания изображений набора MSTAR, это обстоятельство не учитывается, результаты, изложенные в них, по-видимому, являются практически достоверными.

Следует отметить, что задача поиска архитектуры сети, обеспечивающей максимизацию точности классификации, в работе не ставилась. Выбранная архитектура сети C-CNN соответствует уже классической на сегодняшний день архитектуре сети VGG-типа, и обеспечивает получение оценок точности классификации, по-видимому, характерных для ГСНС вообще. Это подтверждается тем, что использование предобученных сетей с большим количеством слоев и более сложной архитектурой в рассматриваемой задаче классификации РЛИ набора MSTAR не дает заметных преимуществ. Более того, во всех случаях они продемонстрировали более низкую точность классификации. Это можно объяснить однородностью наборов данных и их небольшим объемом, что не позволило рассмотренным сетям реализовать преимущества большей глубины и сложности архитектуры по сравнению с C-CNN. Кроме того, для случая монохромных радиолокационных изображений они являются избыточными, так как обучены на трехканальных цветных изображениях набора ImageNet.

Литература

1. Alzubaidi L., Zhang J., Humaidi A.J., Al-Dujaili A., Duan Y., Al-Shamma O., Santamaria J., Fadhel M.A., Al-Amidie M., Farhan L. Review Of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions. *Journal of Big Data*. 2021. V.8. №53.
2. Rawat W., Wang Z. Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. *Neural Computation*. 2017. V.29. P.2352-2449. https://doi.org/10.1162/neco_a_00990
3. Созыкин А.В. Обзор методов обучения глубоких нейронных сетей. *Вестник ЮУрГУ. Серия «Вычислительная математика и информатика»*. 2017. Т.6. №3. С.28-59. <https://doi.org/10.14529/cmse170303>

4. Zhu X., Montazeri S., Ali M., Hua Yu., Wang Yu., Mou L., Shi Yi., Xu F., Bamler R. Deep Learning Meets SAR. *ArXiv*. <https://arxiv.org/abs/2006.10027>
5. Wang P., Patel V.M. Generating High Quality Visible Images from SAR Images Using CNNs. *ArXiv*. 2018. <https://arxiv.org/abs/1802.10036>
6. Rittenbach A., Walters J.P. RDAnet: A Deep Learning Based Approach for Synthetic Aperture Radar Image Formation. *ArXiv*. 2020. <https://arxiv.org/abs/2001.08202>
7. Hu C., Wang L., Li Z., Zhu D. Inverse Synthetic Aperture Radar Imaging Using a Fully Convolutional Neural Network. *IEEE Geoscience and Remote Sensing Letters*. 2020. V.17. №7. P.1203-1207. <https://doi.org/10.1109/LGRS.2019.2943069>
8. Wang H., Chen S., Xu F., Jin Y.-Q. Application of Deep-Learning Algorithms to MSTAR Data. *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*. 2015. P.3743-3745. <https://doi.org/10.1109/IGARSS.2015.7326637>
9. Chen S., Wang H., Xu F., Jin Y.-Q. Target Classification Using the Deep Convolutional Networks for SAR Images. *IEEE Transaction Geoscience and Remote Sensing*. 2016. V.54. №8. P.4806-4817. <https://doi.org/10.1109/TGRS.2016.2551720>
10. Anas H., Majdoulayne H., Chaimae A., Nabil S.M. Deep Learning for SAR Image Classification. *Intelligent Systems and Applications*. 2020. P.890-898. https://doi.org/10.1007/978-3-030-29516-5_67
11. Chen S., Wang H. SAR Target Recognition Based on Deep Learning. *2014 International Conference on Data Science and Advanced Analytics (DSAA)*. 2014. P.541-547. <https://doi.org/10.1109/DSAA.2014.7058124>
12. Coman C., Thaens R. A Deep Learning SAR Target Classification Experiment on MSTAR Dataset. *2018 19th International Radar Symposium (IRS)*. 2018. P.1-6. <https://doi.org/10.23919/IRS.2018.8448048>
13. Furukawa H. Deep Learning for End-to-End Automatic Target Recognition from Synthetic Aperture Radar Imagery. *ArXiv*. <https://arxiv.org/abs/1801.08558>

14. Profeta A., Rodriguez A., Clouse H.S. Convolutional Neural Networks for Synthetic Aperture Radar Classification. *Proc. SPIE 9843, Algorithms for Synthetic Aperture Radar Imagery XXIII, 98430M*. <https://doi.org/10.1117/12.2225934>
15. Wang Z., Xu X. Efficient deep convolutional neural networks using CReLU for ATR with limited SAR images. *The Journal of Engineering*. 2019. V.2019. №21. P.7615-7618. <https://doi.org/10.1049/joe.2019.0567>
16. Wilmanski M., Kreucher C., Lauer J. Modern Approaches in Deep Learning for SAR ATR. *Proc. SPIE 9843, Algorithms for Synthetic Aperture Radar Imagery XXIII, 98430N*. <https://doi.org/10.1117/12.2220290>
17. Xie Yi., Dai W., Hu Z., Liu Yi., Li C., Pu X. A Novel Convolutional Neural Network Architecture for SAR Target Recognition. *Journal of Sensors*. 2019. <https://doi.org/10.1155/2019/1246548>
18. Xinyan F., Weigang Z. Research on SAR Image Target Recognition Based on Convolutional Neural Network. *Journal of Physics: Conference Series*. 2019. V.1213(2019) 042019. <https://doi.org/10.1088/1742-6596/1213/4/042019>
19. Zhai J., Dong G., Chen F., Xie X., Qi C., Li L. A Deep Learning Fusion Recognition Method Based On SAR Image Data. *2018 International Conference on Identification, Information and Knowledge in the Internet of Things. Procedia Computer Science*. 2019. V.147. P.533-541. <https://doi.org/10.1016/j.procs.2019.01.229>
20. LeCun Y., Bottou L., Bengio Y., Haffner P. Gradient-Based Learning Applied to Document Recognition. *Proceedings Of The IEEE*. 1998. V.86. №11. P.2278-2324. <https://doi.org/10.1109/5.726791>
21. Krizhevsky A., Sutskever I., Hinton G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Communications of the ACM*. 2017. V.60. №6. P.84-90. <https://doi.org/10.1145/3065386>
22. Gao F., Huang T., Sun J., Wang J., Hussain A., Yang E. A New Algorithm of SAR Image Target Recognition based on Improved Deep Convolutional Neural Network. *Cognitive Computation*. 2019. V.11. P.809-824. <https://doi.org/10.1007/s12559-018-9563-z>

23. Malmgren-Hansen D., Engholm R., Østergaard Pedersen M. Training Convolutional Neural Networks for Translational Invariance on SAR ATR. *Proceedings of EUSAR 2016: 11th European Conference on Synthetic Aperture Radar*. 2016. P.459-462.
24. Бородинов А.А., Мясников В.В. Сравнение алгоритмов классификации радарных изображений при различных методах предобработки на примере базы MSTAR. *Сборник трудов IV международной конференции и молодежной школы «Информационные технологии и нанотехнологии» (ИТНТ-2018)*. Самара, Новая техника. 2018. С.586-594.
25. Татузов А.Л. *Нейронные сети в задачах радиолокации*. Кн. 28. Москва, Радиотехника. 2009. 432 с.
26. Биленко С.В., Чередеев К. Ю., Зограбян М. К. Перспективы использования глубоких нейронных сетей в радиолокации. *Вопросы радиоэлектроники*. 2017. №1. С.57-63.
27. Казачков Е.А., Матюгин С.Н., Попов И.В., Шаронов В.В. Обнаружение и классификация малоразмерных объектов на изображениях, полученных радиолокационными станциями с синтезированной апертурой. *Вестник Концерна ВКО «Алмаз - Антей»*. 2018. №1. С.93-99.
28. Чорбаа Н.А., Ле Ань Ту, Толстой И.М. Сравнительный анализ методов детектирования объектов на радиолокационных изображениях при помощи нейронных сетей. Научный результат. Информационные технологии. 2020. Т.5. №4. С.15-25. <https://doi.org/10.18413/2518-1092-2020-5-4-0-3>
29. Kechagias-Stamatis O., Aouf N. Automatic Target Recognition on Synthetic Aperture Radar Imagery: A Survey. *ArXiv*. <https://arxiv.org/abs/2007.02106>
30. Шолле Ф. *Глубокое обучение на Python*. Санкт-Петербург, Питер. 2018. 400 с.
31. Николенко С., Кадурич А., Архангельская Е. *Глубокое обучение*. Санкт-Петербург, Питер. 2018. 480 с.
32. Agarwal T., Sugavanam N., Ertin E. Sparse Signal Models for Data Augmentation in Deep Learning ATR. *ArXiv*. <https://arxiv.org/abs/2012.09284>

33. Furukawa H. Deep Learning for Target Classification from SAR Imagery Data Augmentation and Translation Invariance. *ArXiv*. <https://arxiv.org/abs/1708.07920>
34. Malmgren-Hansen D., Kusk A., Dall J., Nielsen A. A., Engholm R., Skriver H. Improving SAR Automatic Target Recognition Models with Transfer Learning from Simulated Data. *IEEE Geoscience and Remote Sensing Letters*. 2017. V.14(9). P.1484-1488. <https://doi.org/10.1109/LGRS.2017.2717486>
35. Wang J., Virtue P., Yu S.X. Joint Embedding and Classification for SAR Target Recognition. *ArXiv*. <https://arxiv.org/abs/1712.01511>
36. Simonyan K., Zisserman A. Very Deep Convolutional Networks For Large-Scale Image Recognition. *ArXiv*. <https://arxiv.org/abs/1409.1556>
37. Huang Z., Pan Z., Lei B. What, Where and How to Transfer in SAR Target Recognition Based on Deep CNNs. *ArXiv*. <https://arxiv.org/abs/1906.01379>
38. Szegedy C., Liu W., Jia Ya., Sermanet P., Reed S., Anguelov D., Erhan D., Vanhoucke V., Rabinovic A. Going deeper with convolutions. *ArXiv*. <https://arxiv.org/abs/1409.4842>
39. Szegedy C., Vanhoucke V., Ioffe S., Shlens J., Wojna Z. Rethinking the Inception Architecture for Computer Vision. *ArXiv*. <https://arxiv.org/abs/1512.00567>
40. He K., Zhang X., Ren S., Sun J. Deep Residual Learning for Image Recognition. *ArXiv*. <https://arxiv.org/abs/1512.03385>
41. Szegedy C., Vanhoucke V., Ioffe S. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *ArXiv*. <https://arxiv.org/abs/1602.07261>
42. Chollet F. Xception: Deep Learning with Depthwise Separable Convolutions. *ArXiv*. <https://arxiv.org/abs/1610.02357>
43. Huang G., Liu Z., van der Maaten L. Densely Connected Convolutional Networks. *ArXiv*. <https://arxiv.org/abs/1608.06993>
44. Howard A.G., Zhu M., Chen B., Kalenichenko D. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *ArXiv*. <https://arxiv.org/abs/1704.04861>

Для цитирования:

Купряшкин И.Ф. Сравнительные результаты точности классификации радиолокационных изображений объектов набора MSTAR сверточными нейронными сетями с различными архитектурами. *Журнал радиоэлектроники* [электронный журнал]. 2021. №11. <https://doi.org/10.30898/1684-1719.2021.11.14>