

УДК 004.932

РАЗРАБОТКА СИСТЕМЫ РАСПОЗНАВАНИЯ ДОРОЖНЫХ ЗНАКОВ НА ОСНОВЕ ОДНОСТУПЕНЧАТОГО ДЕТЕКТОРА

Чулин К.В., Белов Ю.С.

ФГБОУ ВО «Московский государственный технический университет имени Н.Э. Баумана»,
филиал, Калуга, e-mail: fn1-kf@mail.ru

Обнаружение дорожных знаков является чрезвычайно важным в системах автономного вождения и в целом влияет на безопасность на дороге. Для снижения риска несчастных случаев и улучшения опыта вождения водителей были разработаны системы распознавания дорожных знаков, которые сыграли важную роль в автоматическом вождении и обслуживании дорожной сети. В данной статье предложена новая модель, основанная на сверточной нейронной сети, для повышения точности обнаружения и распознавания дорожных знаков, особенно в условиях плохой видимости и крайне ограниченного зрения. Общая модель системы распознавания дорожных знаков состоит из ряда взаимосвязанных модулей. Для повышения эффективности вычислений был предложен одноступенчатый детектор. Система состоит из основной сети, вспомогательной сети и детектора. Основная сеть извлекает объекты из изображения и захватывает объекты. С начальными выходными функциями основной сети вспомогательная сеть объединяет функции и адаптирует размер, тем самым улучшая общую производительность архитектуры. Детектор получает три выходных сигнала от сети и выдает прогноз положения ограничивающей рамки, достоверности объекта и классов объектов для каждого выходного слоя карты объектов. С помощью высококачественных карт объектов каждого слоя обнаружение может быть более точным.

Ключевые слова: обнаружение дорожных знаков, распознавание дорожных знаков, изображения, классификация, смешанная свертка

DEVELOPMENT OF A ROAD SIGN RECOGNITION SYSTEM BASED ON A SINGLE-STAGE DETECTOR

Chulin K.V., Belov Yu.S.

Bauman Moscow State Technical University, Kaluga branch, Kaluga, e-mail: fn1-kf@mail.ru

The detection of road signs is extremely important in autonomous driving systems and a generally influential aspect of road safety. To reduce the risk of accidents and improve the driving experience of drivers, road sign recognition systems have been developed, which have played an important role in automatic driving and maintenance of the road network. This article proposes a new model based on a convolutional neural network to improve the accuracy of detection and recognition of road signs, especially in conditions of poor visibility and extremely limited vision. The general model of the road sign recognition system consists of a number of interconnected modules. To increase the efficiency of calculations, a single-stage detector was proposed. The system consists of a main network, an auxiliary network and a detector. The main network extracts objects from the image and captures objects. With the initial output functions of the main network, the auxiliary network combines the functions and adapts the size, thereby improving the overall performance of the architecture. The detector receives three output signals from the network and provides a forecast of the position of the bounding box, the reliability of the object and object classes for each output layer of the object map. With high-quality feature maps of each layer, detection can be more accurate.

Keywords: traffic sign detection, traffic sign recognition, images, classification, mix convolution

Обнаружение дорожных знаков чрезвычайно важно в системах автономного вождения и безопасности на транспорте. Однако точное распознавание дорожных знаков остается сложной задачей, особенно в экстремальных условиях. В суровых погодных условиях, таких как туман и снег, дорожно-транспортные происшествия часто происходят из-за рассеянного вождения, невнимательности или плохой видимости. Для снижения риска несчастных случаев и улучшения опыта вождения водителей были разработаны системы распознавания дорожных знаков, которые сыграли важную роль в автоматическом вождении и обслуживании дорожной сети. Системы распознавания дорожных знаков можно разделить на две подзадачи, т.е. обнаружение и клас-

сификацию, первая из которых направлена на идентификацию целевых объектов по изображениям, а вторая направлена на классификацию обнаруженных дорожных знаков по подклассам [1].

Цель исследования – разработать систему распознавания дорожных знаков, основанную на сверточной нейронной сети, для повышения точности обнаружения и распознавания дорожных знаков.

Описание модели. Предлагается новая модель, основанная на сверточной нейронной сети, для повышения точности обнаружения и распознавания дорожных знаков, особенно в условиях плохой видимости и крайне ограниченного зрения. Общая модель системы распознавания дорожных знаков состоит из ряда взаимосвязанных модулей.

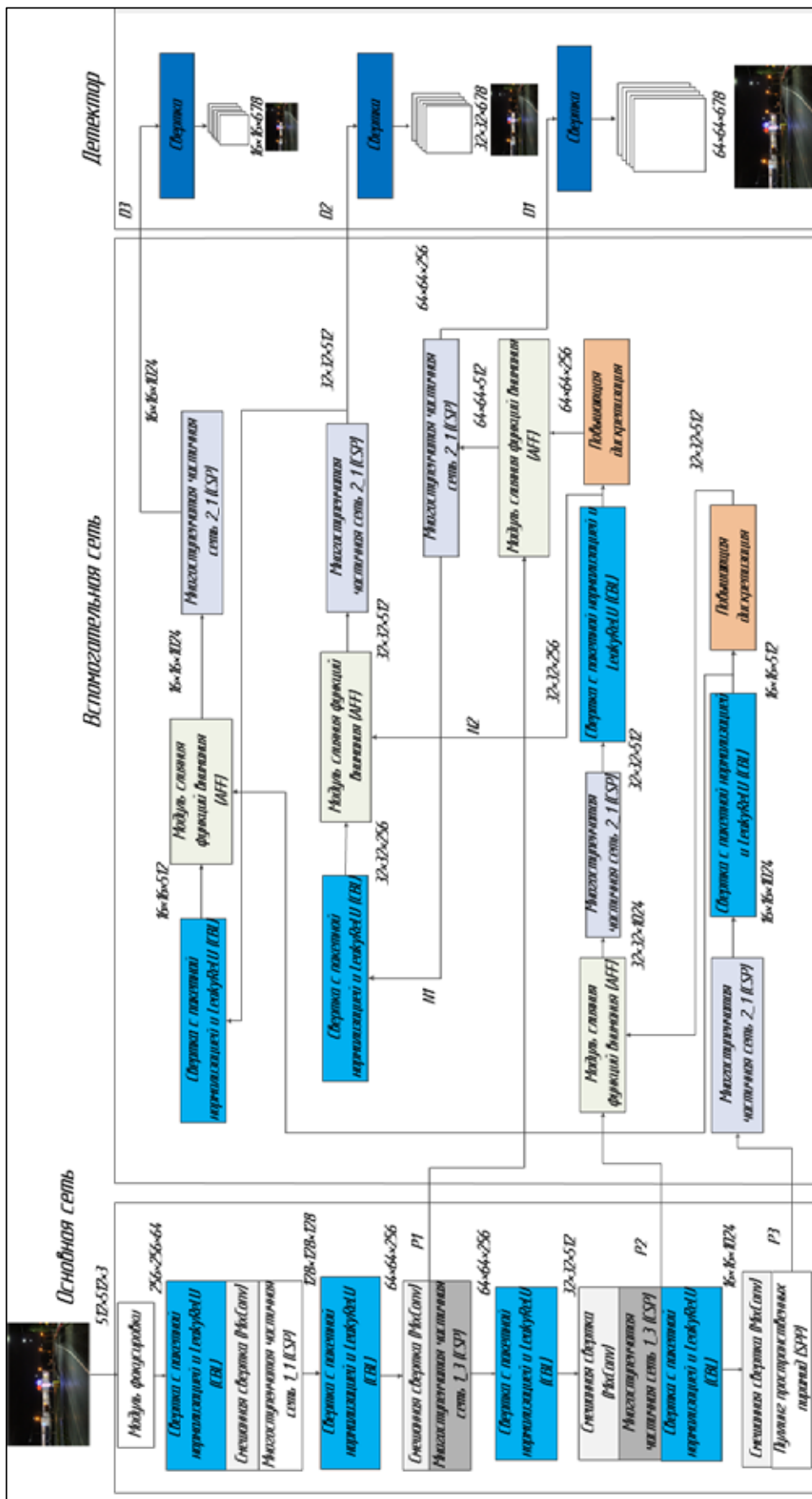


Рис. 1. Общая архитектура предлагаемого подхода

Для повышения эффективности вычислений был предложен одноступенчатый детектор. Одноступенчатый детектор не имеет этапа регионального предложения; вместо этого он предсказывает класс и местоположение объекта и получает конечный результат с помощью одного этапа. Наиболее представительными моделями являются YOLO и SSD. Одноступенчатая модель распознавания дорожных знаков обладает высокой эффективностью, но недостаточной точностью [2].

Общая архитектура модели. Общая архитектура показана на рис. 1. Система состоит из основной сети, вспомогательной сети и детектора. Для того чтобы определить местоположение и класс объекта, необходимо извлечь объекты из изображения и захватить объекты с использованием основной сети для позиционирования и классификации. Была использована смешанная свертка, которая могла выполнять сверточную операцию со смешанными размерами ядра, так что можно было захватывать различные шаблоны с различными разрешениями. С начальными выходными функциями основной сети вспомогательная сеть объединяет функции и адаптирует размер, тем самым улучшая общую производительность архитектуры. Модуль слияния функций внимания (AFF – attentional feature fusion) использовался во вспомогательном слое для объединения функций, полученных из одного и того же слоя или перекрестного слоя, на основе внимания, включая короткие и длинные соединения с пропуском, и даже выполнял начальное слияние внутри себя. Детектор

получает три выходных сигнала от сети и выдает прогноз положения ограничивающей рамки, достоверности объекта и классов объектов для каждого выходного слоя карты объектов. С помощью высококачественных карт объектов каждого слоя обнаружение может быть более точным [3].

Основная сеть для извлечения объектов из изображения. Рассмотрим основную сеть. Основная сеть состояла из модуля фокусировки, свертки с пакетной нормализацией и LeakyReLU (CBL – Convolution with Batch normalization and LeakyRelu), смешанной свертки (MixConv – mix convolution), многоступенчатой частичной сети (CSP – Cross Stage Partial Network) и пуллинга пространственных пирамид (SPP – Spatial Pyramid Pooling). Размер входного изображения составлял $512 \times 512 \times 3$, среди которых 512×512 представляли ширину и высоту в пикселях соответственно, а 3 представляли 3 канала. После модуля фокусировки размер был изменен на $256 \times 256 \times 64$. После следующего модуля свертки с пакетной нормализацией и LeakyReLU (CBL – Convolution with Batch normalization and LeakyRelu) размер стал $128 \times 128 \times 128$.

Модуль фокусировки. Модуль фокусировки изображен на рис. 2.

Модуль фокусировки взял входное изображение размером $512 \times 512 \times 3$ и выполнил операцию нарезки. Эта операция извлекла каждый второй пиксель из изображения аналогично понижающей дискретизации. Затем из входного изображения были получены 4 изображения, 4 изображения дополняли друг друга, и, таким образом, отсутствовали данные.

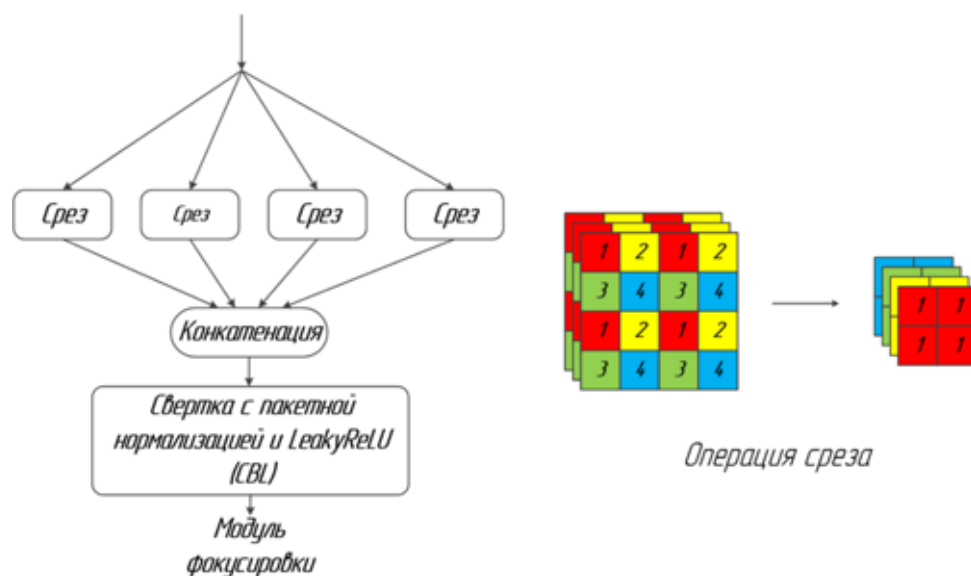


Рис. 2. Модуль фокусировки

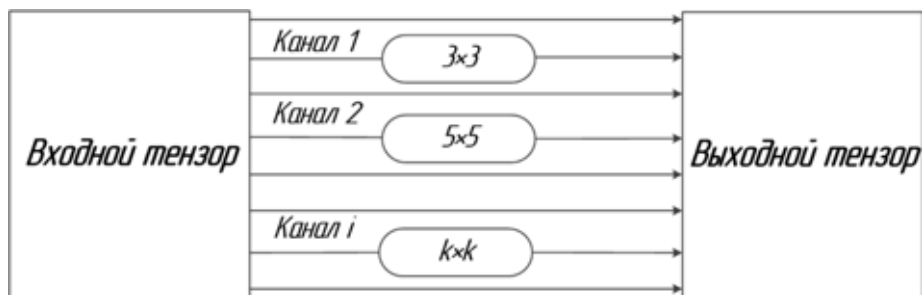


Рис. 3. Смешанная свертка (MixConv – mix convolution)

После этого информация о ширине и информация о высоте были объединены в пространство канала, а входное пространство было расширено в 4 раза. Таким образом, объединенное изображение имело 12 каналов, в то время как исходное изображение имело только 3 канала. Наконец, новое изображение было свернуто для получения двух карт объектов с понижающей дискретизацией без каких-либо отсутствующих данных. В этом случае была получена карта объектов размером $256 \times 256 \times 12$.

Как показано в правом нижнем углу на рис. 2, после операции разрезания изображения изображение размером $4 \times 4 \times 3$ было изменено на карту объектов размером $2 \times 2 \times 12$. После свертки с каналом 64 вывод с размером $256 \times 256 \times 64$ было получено. Модуль фокусировки может помочь уменьшить вычислительную нагрузку при понижающей дискретизации, но он не приведет к отсутствию каких-либо данных. Таким образом, модуль фокусировки может сохранять более полные данные с уменьшением изображения для последующего извлечения объектов [4].

Модуль смешанной свертки. Следующий модуль смешанной свертки (MixConv – mix convolution) (рис. 3) и модуль CSP1_1 не изменили размер, и размер изображения по-прежнему составлял $128 \times 128 \times 128$ после этих двух модулей.

Смешанная свертка (MixConv – mix convolution) смешала различные размеры ядра (3×3 , 5×5 и 7×7) в одной операции свертки, так что различные шаблоны с различными разрешениями могут быть легко зафиксированы.

Входной тензор имеет форму h, w, c и обозначается как $X^{(h, w, c)}$,

где h – высота,
 w – ширина,
 c – размер канала.

Сверточное ядро по глубине может быть обозначено как $W^{(k, k, c, m)}$, где $k \times k$ относится к размеру ядра, c относится к размеру

входного канала, а m относится к множителю канала.

Чтобы упростить анализ, предполагаем, что ядро имеет одинаковую ширину и высоту (k). С учетом этого предположения для выходного тензора $Y^{(h, w, c^*m)}$ были получены одинаковые формы h, w и умноженный размер выходного канала $m \cdot c$. Карта характеристик каждого выходного сигнала может быть рассчитана с помощью формулы

$$Y_{x, y, z}^t = \sum_{\substack{k_t \leq x \leq k_t \\ \frac{k_t}{2} \leq y \leq \frac{k_t}{2}}} \widehat{X}_{x+i, y+j, \frac{z}{m}}^t \cdot W_{i, j, z}^t, \quad \forall z=1, \dots, m \cdot c_t. \quad (1)$$

Как показано на рис. 5, каналы были разделены на несколько групп с помощью смешанной свертки, и разные размеры ядра были применены к разным группам. Более конкретно, входной тензор был разделен на g групп виртуальных тензоров, т.е. $\langle X \wedge^{(h, w, c_1)}, \dots, X \wedge^{(h, w, c_g)} \rangle$.

Пространственная высота и ширина всех виртуальных тензоров были одинаковыми, то есть h и w соответственно. Кроме того, общий размер канала этих виртуальных тензоров был таким же, как и у исходного входного тензора, т.е. $c_1 + c_2 + \dots + c_g = c$. Аналогично, сверточное ядро также было разделено на g групп виртуальных ядер $\langle W \wedge^{(k_1, k_1, c_1, m)}, \dots, W \wedge^{(k_g, k_g, c_g, m)} \rangle$.

Соответствующий виртуальный выходной сигнал для t -й группы виртуальных входных тензоров и ядер может быть вычислен по уравнению

$$Y_{x, y, z}^t = \sum_{\substack{k_t \leq x \leq k_t \\ \frac{k_t}{2} \leq y \leq \frac{k_t}{2}}} \widehat{X}_{x+i, y+j, \frac{z}{m}}^t \cdot W_{i, j, z}^t, \quad \forall z=1, \dots, m \cdot c_t. \quad (2)$$



Рис. 4. Свертка с пакетной нормализацией и LeakyReLU (CBL)

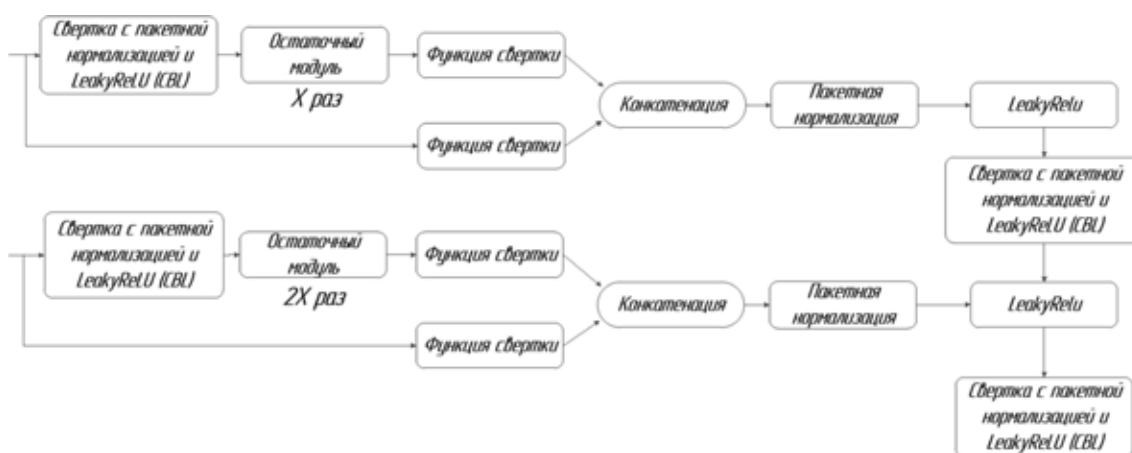


Рис. 5. (а) Многоступенчатая частичная сеть (CSP – Cross Stage Partial Network) 1_X,
(б) Многоступенчатая частичная сеть (CSP) 2_X

Конечный выходной тензор может быть получен путем объединения всех виртуальных выходных тензоров:

$$Y_{x,y,z_0} = \text{Concat}\left(\widehat{Y_{x,y,z_1}^1}, \dots, \widehat{Y_{x,y,z_g}^g}\right). \quad (3)$$

Размер канала конечного выходного сигнала равен $z_0 = z_1 + \dots + z_g = m \cdot c$.

Смешанная свертка (MixConv – mix convolution) смешивала различные размеры ядра (3×3 , 5×5 и 7×7) в одной операции свертки, чтобы можно было легко захватывать различные шаблоны с различными разрешениями [5].

Модуль свертки с пакетной нормализацией и LeakyReLU. Следующим шагом был модуль свертки с пакетной нормализацией и LeakyReLU (CBL – Convolution with Batch normalization and LeakyRelu) (рис. 4), который изменил размер изображения на $64 \times 64 \times 256$. Затем модуль свертки не внес никаких изменений в изображение, и размер по-прежнему составлял $64 \times 64 \times 256$. После этого модуль CSP1_3 не изменил размер вывода. Таким образом, выходной сигнал размером $64 \times 64 \times 256$ был записан как P1.

Модуль свертки с пакетной нормализацией и LeakyReLU (CBL – Convolution with Batch normalization and LeakyRelu) в основной сети состоял из функции свертки, пакетной нормализации и LeakyReLU. Остаточный блок использовался в многоступенчатой частич-

ной сети (CSP – Cross Stage Partial Network). Как показано на рис. 5, б, остаточный модуль состоял из двух блоков CBL, которые были подключены непрерывно. Исходный ввод и вывод 2-го модуля свертки с пакетной нормализацией и LeakyReLU (CBL – Convolution with Batch normalization and LeakyRelu) выполняли функцию сложения векторов в качестве выходных данных [6].

Модуль многоступенчатой частичной сети. CSP1_X изображен на рис. 5 в верхней части. Исходный ввод CSP1 проходил через модуль свертки с пакетной нормализацией и LeakyReLU (CBL – Convolution with Batch normalization and LeakyRelu) и X остаточных модулей (ResUnit) (рис. 6). В конце концов он выполнил сверточную функцию для получения временного вывода основного пути. Тем временем исходный ввод выполнил другую сверточную функцию, прошел по другому пути, а затем объединился с выводом основного пути. Результат прошел через нормализацию пакета, LeakyReLU и свертки с пакетной нормализацией и LeakyReLU (CBL – Convolution with Batch normalization and LeakyRelu).

CSP2_X описан на рис. 5 в нижней части, и его структура немного отличается от CSP1_X. В CSP2_X основной путь состоял из $2 \cdot X$ кратных единиц свертки с пакетной нормализацией и LeakyReLU (CBL – Convolution with Batch normalization and LeakyRelu) вместо повторного объединения в CSP1_X.



Рис. 6. Остаточный модуль (ResUnit)

Исходный ввод CSP2_X прошел через $2 \times X$ единицы CBL, а затем сверточную функцию, чтобы получить временный вывод основного пути. Тем временем исходный ввод выполнил другую сверточную функцию, прошел по другому пути, а затем объединился с выводом основного пути. Результат прошел через пакетную нормализацию, LeakyReLU.

В деталях CSP улучшает обучаемость сверточной нейронной сети, сокращает вычислительную сложность, обеспечивает высокую точность и малый вес, а также снижает стоимость памяти. CSP может интегрировать изменения градиента в карту объектов от начала до конца, что может снизить вычислительные затраты при обеспечении точности.

Следующим шагом был модуль CBL, который изменил размер выходного сигнала на $32 \times 32 \times 512$. Затем модуль MixConv не внес никаких изменений в размер, и модуль CSP1_3 на следующем шаге также не изменил размер. Вывод здесь размером $32 \times 32 \times 512$ был записан как P2.

Следующим шагом был модуль CBL, который изменил размер на $16 \times 16 \times 1024$. Затем модуль MixConv и модуль пуллинга пространственных пирамид (SPP – Spatial Pyramid Pooling) на следующем шаге не изменили размер вывода. Вывод здесь размером $16 \times 16 \times 1024$ был записан как P3 [7].

Заключение

В данной статье продемонстрирована система распознавания дорожных знаков распознавания дорожных знаков, основанная на сверточной нейронной сети, для повышения точности обнаружения и распознавания дорожных знаков [8].

Список литературы

1. Dhar P., Abedin Md.Z., Biswas T. Traffic sign detection – A new approach and recognition using convolution neural network. IEEE Region 10 Humanitarian Technology Conference. 2017. P. 416–419.
2. Смольянинов В.А., Белов Ю.С. Проектирование программного комплекса обнаружения и распознавания дорожных знаков в потоковом видео // Научное обозрение. Технические науки. 2021. № 4. С. 16–21.
3. Pei S., Tang F., Ji Y., Fan J., Ning Z. Localized traffic sign detection with multi-scale deconvolution networks. IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC). 2018. P. 355–360.
4. Belghaouti O., Handouzi W., Tabaa M. Improved traffic sign recognition using deep ConvNet architecture. Procedia Computer Science. 2020. Vol. 177. P. 468–473.
5. Saadna Y., Behloul A. An overview of traffic sign detection and classification methods. International Journal of Multimedia Information Retrieval 6. 2017. P. 193–210.
6. Аккуратов В.В., Белов Ю.С. Выбор архитектуры приложения для задачи распознавания дорожных знаков // Наука, техника и образование. 2018. № 3 (21). С. 26–34.
7. Tabernik D., Skocaj D. Deep learning for large-scale traffic-sign detection and recognition. IEEE Transactions on Intelligent Transportation Systems. 2020. Vol. 21, No. 4. P. 1427–1440.
8. Sheikh M.A.A., Kole A., Maity T. Traffic sign detection and classification using colour feature and neural network. Conference on Intelligent Control Power and Instrumentation (ICICPI). 2016. P. 307–311.