

Р. М. Хусаинов

ЧИСЛЕННЫЙ МЕТОД ИНТЕГРАЦИИ СЛОЕВ В АРХИТЕКТУРУ АДАПТИВНОЙ ЭНЕРГОЭФФЕКТИВНОЙ МОДЕЛИ

Ключевые слова: адаптивная энергоэффективная модель, метод интеграции, модификация, точность распознавания, параметризованный фильтр, рассеивающий фильтр, объект, энергоэффективность.

В данной статье рассматривается задача распознавания объектов в дорожных сценах. В процессе исследования выявлено, что увеличение сложности нейросетевых моделей приводит к значительному росту вычислительных затрат и энергопотребления, что ограничивает их применение в мобильных устройствах. Для решения задачи предлагается метод интеграции слоев в архитектуру адаптивной энергоэффективной модели, отличающийся динамическим управлением параметрами модели на основе анализа дорожной сцены, вычислительных ресурсов, что позволяет повысить точность распознавания объектов (дорожных знаков, дорожных разметок, светофоров) в дорожных сценах с учетом естественных шумов (дождь, солнечный блик, туман, снег, сумерки) и энергоэффективность моделей. Разработанный метод имеет преимущество по сравнению со стандартными моделями YOLOv8, обеспечивая оптимальный баланс между точностью распознавания и энергопотреблением, что особенно важно для устройств с ограниченными вычислительными ресурсами (смартфонами, планшетами). С учетом добавления слоев устойчивости и оптимизации адаптивные энергоэффективные модели имеют модификации: YOLOv8n-AE, YOLOv8s-AE, YOLOv8m-AE, YOLOv8l-AE, YOLOv8x-AE. Проведен пример численного расчета параметров слоев «Adaptability» и «Optimization» дорожной сцены. Реализация метода интеграции слоев в архитектуру адаптивной энергоэффективной модели выполнена в прикладной программе, что позволяет снизить нагрузку на вычислительные ресурсы для использования в мобильных устройствах, обеспечивая точный анализ дорожной сцены. Разработанные модели используются в мобильном приложении «Ассистент водителя» для распознавания объектов в режиме реального времени, обеспечивая надежную работу вычислительных ресурсов. В перспективе целесообразно усовершенствование прикладной программы, расширение базы данных для дообучения предложенных моделей.

R. M. Khusainov

NUMERICAL METHOD FOR INTEGRATING LAYERS INTO THE ARCHITECTURE OF AN ADAPTIVE ENERGY-EFFICIENT MODEL

Keywords: adaptive energy-efficient model, integration method, modification, recognition accuracy, parameterized filter, scattering filter, object, energy efficiency.

This article considers the problem of object recognition in road scenes. The study revealed that the increase in the complexity of neural network models leads to a significant increase in computational costs and energy consumption, which limits their use in mobile devices. To solve the problem, a method is proposed for integrating layers into the architecture of an adaptive energy-efficient model, characterized by dynamic control of the model parameters based on the analysis of the road scene, computing resources, which allows to increase the accuracy of object recognition (road signs, road markings, traffic lights) in road scenes taking into account natural noise (rain, sun glare, fog, snow, twilight) and the energy efficiency of models. The developed method has an advantage over standard YOLOv8 models, providing an optimal balance between recognition accuracy and energy consumption, which is especially important for devices with limited computing resources (smartphones, tablets). Taking into account the addition of stability and optimization layers, adaptive energy-efficient models have modifications: YOLOv8n-AE, YOLOv8s-AE, YOLOv8m-AE, YOLOv8l-AE, YOLOv8x-AE. An example of numerical calculation of the parameters of the «Adaptability» and «Optimization» layers of the road scene is given. The implementation of the method of integrating layers into the architecture of the adaptive energy-efficient model is performed in the application program, which reduces the load on computing resources for use in mobile devices, ensuring accurate analysis of the road scene. The developed models are used in the mobile application «Driver Assistant» for object recognition in real time, ensuring reliable operation of computing resources. In the future, it is advisable to improve the application program, expand the database for additional training of the proposed models.

Введение

Современные системы компьютерного зрения, применяемые в интеллектуальных транспортных средствах и системах помощи водителю Advanced Driver-Assistance Systems (ADAS), требуют высокой точности распознавания объектов в дорожных сценах [1-3]. Увеличение сложности нейросетевых моделей приводит к значительному росту вычислительных затрат и энергопотребления, что ограничивает их применение в мобильных устройствах: смартфонах, планшетах [4, 5].

Актуальной задачей является разработка архитектуры адаптивной энергоэффективной

модели, способной динамически оптимизировать структуру в зависимости от сложности входных данных. В данной работе предлагается численный метод интеграции слоев в архитектуру данной модели, отличающийся динамическим управлением параметрами модели на основе анализа дорожной сцены, вычислительных ресурсов, что позволяет повысить точность распознавания объектов дорожной инфраструктуры и энергоэффективность моделей.

Постановка задачи

Приведена постановка задачи по построению

моделей распознавания объектов. Существует архитектура YOLOv8, которая разделена на три основных компонента: Backbone (Позвоночник), Neck (Шея) и Head (Голова). Каждая из перечисленных компонентов имеет свои особенности.

$I = \{I_1, I_2, \dots, I_N\}$ представляет собой множество дорожных сцен, на которых изображены знаки дорожного движения, дорожные разметки и светофоры размерностью N . $O = \{O_1, O_2, \dots, O_i\}$ – множество объектов, каждый из которых характеризуется координатами ограничивающей рамки (x_i, y_i, w_i, h_i), классом c_i и точностью распознавания p_i .

Каждая дорожная сцена сформирована с учетом естественных шумов (дождь, снег, солнечный блик, туман, сумерки).

Для модификации модели YOLOv8 требуется разработать метод интеграции слоев «Adaptability» и «Optimization» в компоненты архитектуры: Backbone (Позвоночник) и Head (Голова). В слое «Adaptability» необходимо привести формулы расчета параметров фильтров, в слое «Optimization» – формулы расчета параметров оптимизации модели. Для оценки эффективности модели необходимо выполнить численный расчет параметров слоев в дорожной сцене.

Построение архитектуры адаптивной энергоэффективной модели

Адаптивные энергоэффективные модели формируются с учетом критериев, влияющих на их производительность в зависимости от решаемой задачи. Формула 1 отражает обобщенный критерий эффективности модели, который зависит от пяти параметров: архитектуры, метрик классификации, времени обработки одного изображения, размера модели и энергоэффективности [6].

Эффективность модели в контексте построения архитектуры адаптивной энергоэффективной системы подразумевает оптимальное соотношение между метриками классификации, вычислительными затратами и энергопотреблением.

Формула 1 используется для персональных компьютеров и мобильных устройств, где основными параметрами являются ресурсоэффективность и производительность.

$$АЭМ = f(A, M, B, P, Э), \quad (1)$$

где f – функциональная зависимость модели от параметров $A, M, B, P, Э$; A – архитектура модели. Как правило, архитектура бывает следующей: LeNet, AlexNet, VGG16, GoogleNet, ResNet, R-CNN, Fast R-CNN, Faster R-CNN, YOLO, SSD, RetinaNet; M – метрики классификации, необходимые для оценки качества модели (Precision, Recall, F1-score, mAP); B – время, необходимое для обработки одного изображения (кадра), мс; P – размер модели (объем памяти, занимаемый моделью), Мбайт; $Э$ – энергоэффективность (основной метрический показатель, оценивающий, сколько кадров модель способна обработать в секунду, затрачивая 1 Ватт мощности), FPS/Вт.

Энергоэффективность вычисляется по формуле 2:

$$Э = \frac{FPS}{W}, \quad (2)$$

где FPS – величина, обратная времени обработки одного кадра (формула 3); W – потребляемая мощность устройства, Вт.

$$FPS = \frac{1}{t}, \quad (3)$$

где t – время обработки одного изображения, с.

Архитектура адаптивной энергоэффективной модели представлена на рисунке 1.

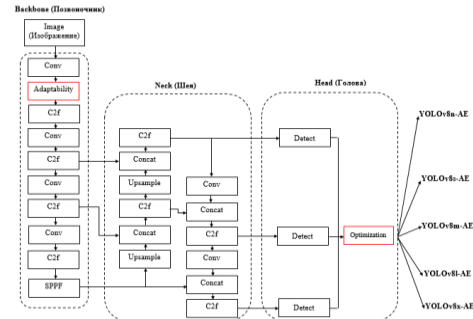


Рис. 1 – Архитектура адаптивной энергоэффективной модели

Fig. 1 – Architecture of an adaptive energy-efficient model

База изображений (дорожных сцен) с размеченными объектами подразделяется на обучающую, валидационную и тестовую выборки, которые передаются на вход нейронной сети [6, 7]. В данной сети уменьшается размер каждого изображения до единого размера (например, 640x640). Архитектура адаптивной энергоэффективной модели разделена на три основных компонента: Backbone (Позвоночник), Neck (Шея) и Head (Голова). Назначение компоненты Backbone (Позвоночник) – извлечение признаков из изображения [8, 9].

Слой «Adaptability» включает адаптивную фильтрацию, шумоподавление и динамическую обработку.

Метод предобработки изображения состоит из параметризованного и рассеивающего фильтров. Параметризованный фильтр – это фильтр, применяемый преобразование к изображению с использованием параметров, зависящих от характеристик данных (яркости, контраста, градиентов), вычисляемый по формуле 4:

$$I_{out}(x, y) = 255 \cdot \left(\frac{I_{in}(x, y)}{255} \right)^{\gamma(x, y)}, \quad (4)$$

где $\gamma(x, y)$ – параметр, вычисляемый на основе локальной статистики (средней яркости μ и стандартного отклонения σ): $\gamma(x, y) = \exp(1 - \frac{\mu(x, y) + \sigma(x, y)}{128})$; $\mu(x, y)$ – средняя яркость (значение интенсивности пикселей в определенной области изображения вокруг точки (x, y)), вычисляемая как: $\mu(x, y) = \frac{1}{N} \sum_{i \in \Omega} I(i)$; $I(i)$ – интенсивность i -го пикселя в окрестности Ω ; N – количество пикселей в окрестности; Ω – заданная область вокруг точки (x, y) ; σ – стандартное отклонение (статистическая мера разброса данных относительно среднего значения), вычисляемая как $\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2}$; x_i – значение пикселя.

С учетом подстановки выражений в формулу (4) параметризованный фильтр вычисляется следующим образом:

$$I_{out}(x, y) = 255 \cdot \left(\frac{I_{in}(x, y)}{255} \right) \exp \left(1 - \frac{\frac{1}{N} \sum_{i \in \Omega} I(i) + \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \frac{1}{N} \sum_{i \in \Omega} I(i))^2}}{128} \right). \quad (5)$$

Рассеивающий фильтр – это фильтр, который сглаживает изображение, сохраняя границы, вычисляемый по формуле 6:

$$\frac{\partial I}{\partial t} = \text{div}(c(|\nabla I|) \cdot \nabla I), \quad (6)$$

где $c(|\nabla I|)$ – функция проводимости, зависящая от градиента: $c(|\nabla I|) = \frac{1}{1 + (\frac{|\nabla I|}{k})^2}$; k – порог

чувствительности к границам, $k = 0, 1$; ∇I – градиент изображения $(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y})$; div – дивергенция

(расхождение), вычисляемая как: $\text{div} = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y}$;

$F = c(|\nabla I|) \cdot \nabla I$ – векторное поле проводимости.

Параметризованный фильтр используется для адаптации модели к дождю, снегу, рассеивающий фильтр – к солнечному блику, к туману. С учетом подстановки выражений в формулу (6) рассеивающий фильтр вычисляется следующим образом:

$$\frac{\partial I}{\partial t} = \frac{\partial}{\partial x} \left(\frac{1}{1 + (\frac{|\nabla I|}{k})^2} \cdot \frac{\partial I}{\partial x} \right) + \frac{\partial}{\partial y} \left(\frac{1}{1 + (\frac{|\nabla I|}{k})^2} \cdot \frac{\partial I}{\partial y} \right). \quad (7)$$

Слой «SPPF» различает объекты (маленькие, средние, большие) в разных масштабах с использованием параллельного применения слоя «Upsample» из компоненты Neck (Шея). На выходе компоненты Backbone (Позвоночник) образуются несколько карт признаков [10-12]. Назначение компоненты Neck (Шея) – объединение признаков разного масштаба для улучшения распознавания объектов на основе слоев «Upsample» и «Concat». Назначение компоненты Head (Голова) – выделение обнаруженных объектов ограничивающей рамкой (bounding box), предсказание классов дорожных объектов в слоях «Detect» с указанием точности их распознавания. Три слоя «Detect» используются для обработки объектов дорожной инфраструктуры разного масштаба [13-15].

В слое «Optimization» уменьшается размер модели, снижаются вычислительные затраты для ее использования на мобильных устройствах (на смартфонах).

Для каждого обнаруженного объекта дорожной инфраструктуры с координатами (x, y, w, h) , порогом уверенности (p) и классом (c) применяется формула 8:

$$Score_{final} = p \cdot \left(1 - \frac{IoU_{overlap}}{\max(IoU_{threshold}, 0.1)} \right) \cdot \alpha(E), \quad (8)$$

где $IoU_{overlap}$ – площадь пересечения с другими ограничивающими рамками (для подавления дубликатов); $IoU_{threshold}$ – площадь объединения; $\alpha(E)$ – энергонезависимый коэффициент (формула 9).

$$\alpha(E) = \begin{cases} 1.0, & \text{если } E \geq 50\%; \\ 0.7 + 0.3 \cdot \frac{E}{50}, & \text{если } E < 50\%. \end{cases} \quad (9)$$

Алгоритм работы слоя «Optimization»:

Шаг 1. Фильтрация ложных срабатываний: исключаются ограничивающие рамки по выражению $Score_{final} < 0.5$.

Шаг 2. Калибровка координат: корректировка границ ограничивающих рамок через экспоненциальное сглаживание (10):

$$x_{new} = x_{pred} + \lambda \cdot (x_{gt_{avg}} - x_{pred}), \quad (10)$$

где x_{pred} – координата ограничивающей рамки; λ – коэффициент сглаживания ($\lambda = 0, 1$), $x_{gt_{avg}}$ – среднее значение по валидационным выборкам.

На выходе компоненты Head (Голова) выводится результат распознавания с указанием классов и точности обнаруженных объектов, а также формируются адаптивные энергоэффективные модели.

С учетом внесения изменений параметров слоев «Adaptability» и «Optimization» в архитектуру адаптивные энергоэффективные модели имеют модификации: YOLOv8n-AE, YOLOv8s-AE, YOLOv8m-AE, YOLOv8l-AE, YOLOv8x-AE.

Таким образом, метод расчета архитектуры адаптивной энергоэффективной модели состоит из последовательных этапов:

Этап 1. Подготовка входных данных (получение на вход сети исходной дорожной сцены, определение сложности данной сцены, изменение размера изображения).

Этап 2. Предобработка изображения (применение параметризованного и рассеивающего фильтров в зависимости от естественного шума на изображении).

Этап 3. Извлечение признаков (компонента Backbone).

$$F_{Backbone} = F_{SPPF}(F_{C2f}(F_{Conv}(F_{Adapt}(I_{resized})), w) + b)), \quad (11)$$

где F_{SPPF} – слой масштабирования объектов, вычисляемый как $F_{SPPF} = SPPF(F_{C2f})$; F_{C2f} – слой комбинирования признаков с нелинейным преобразованием и функцией активацией ReLU, вычисляемый $F_{C2f} = \text{ReLU}(F_{Conv})$; F_{Conv} – сверточный слой, вычисляемый как $F_{Conv} = \sum_{i=-k}^k \sum_{j=-k}^k w(i, j) \cdot I(x + i, y + j) + b$; $w(i, j)$ – ядро свертки (размер $k \times k$); b – вектор смещения; F_{Adapt} – слой устойчивости; $I_{resized}$ – динамическое изменение размера изображения.

Этап 4. Объединение признаков (компонента Neck).

$$F_{Neck} = F_{Upsample}(F_{Concat}), \quad (12)$$

где $F_{Upsample}$ – слой «Upsample», вычисляемый как $F_{Upsample} = \text{Upsample}(F_{in})$; (F_{in}) – входные признаки, который проходят через слой «Upsample»; F_{Concat} – слой «Concat», вычисляемый как $F_{Concat} = \text{Concat}(F_1, F_2, \dots, F_n)$; $F_1, F_2, \dots, F_n$ – признаки, с которыми проводится объединение.

Этап 5. Распознавание объектов и оптимизация (компонента Head).

$$F_{Head} = Score_{final} \cdot \alpha_t, \quad (13)$$

где $Score_{final} = p \cdot \left(1 - \frac{IoU_{overlap}}{\max(IoU_{threshold}, 0.1)} \right) \cdot \alpha(E)$, $IoU_{overlap}$ – площадь пересечения с другими ограничивающими рамками (для подавления дубликатов); $IoU_{threshold}$ – площадь объединения; $\alpha(E)$ – энергонезависимый коэффициент,

вычисляемый как

$$\alpha(E) = \begin{cases} 1.0, & \text{если } E \geq 50\%; \\ 0.7 + 0.3 \cdot \frac{E}{50}, & \text{если } E < 50\%. \end{cases} \quad (14)$$

α_t – коэффициент настройки параметров модели, вычисляемый как:

$$\alpha_t = \eta \cdot (F(S_t, \gamma, \nabla I) \cdot P(S_t) \cdot R(C_t) \cdot \sum_{i=1}^n w_i O_i(I_t)), \quad (15)$$

где η – коэффициент обучения, $\eta = 0.001-0.1$; $F(S_t, \gamma, \nabla I)$ – коэффициент фильтрации, вычисляемый как $F(S_t, \gamma, \nabla I) = \gamma \cdot c(|\nabla I|)$; $P(S_t)$ – коэффициент сложности дорожной сцены, $P(S_t) = 0.3-1.0$; $R(C_t)$ – коэффициент доступности ресурсов, $R(C_t) = 0.1-1.5$; w_i – веса объектов (дорожный знак: $w_i = 1.0$; светофор: $w_i = 0.9$; дорожная разметка: $w_i = 0.7$); $O_i(I_t)$ – коэффициент значимости объекта в дорожной сцене, $O_i(I_t) = 0-1$.

Этап 6. Выход модели.

Обнаруженные объекты имеют параметры: координаты ограничивающей рамки (центр (x, y) , ширина w , высота h); класс (c) и точность (P) .

Формируются адаптивные энергоэффективные модели с указанием их модификации (n-AE, s-AE, m-AE, l-AE, x-AE), размера (Мбайт).

Расчет параметров слоев в архитектуре адаптивной энергоэффективной модели

Для проведения расчета параметров слоев в архитектуре модели использовались портативный компьютер (ноутбук) с техническими характеристиками: процессор Intel Pentium CPU 3825U, частота процессора – 1.9 ГГц, оперативная память – 8 Гб, операционная система – Windows 10 Pro.

Проведен пример численного расчета параметров слоев «Adaptability» и «Optimization» для дорожной сцены, представленной на рисунке 2. Так как дорожная сцена сложная ($O_i = 3$), то увеличивается размер изображения с 640x640 до 960x960.

Как видно из рисунка, в дорожной сцене отображены три объекта (дорожных знака) и естественный шум (солнечный блик), затрудняющий распознаванию этих объектов.



Рис. 2 – Пример дорожной сцены

Fig. 2 – Example of a road scene

В слое «Adaptability» входные значения дорожной сцены:

$$\begin{aligned} &[250, 245, 240] \\ &[255, 250, 235] \text{ (блик: 250-255, знак: 200)} \\ &[240, 230, 220] \end{aligned}$$

Дорожный знак имеет значение пикселя (200), диапазон 250...255 соответствует естественному шуму (солнечному блику).

Средняя яркость μ :

$$\mu = \frac{250+245+240+255+250+235+240+230+220}{9} \approx 240.$$

Стандартное отклонение σ :

$$\sigma = \sqrt{\frac{(250-240)^2 + (245-240)^2 + (240-240)^2 + \dots + (220-240)^2}{9}} \approx 10.8.$$

Параметр γ :

$$\gamma(x, y) = \exp(1 - \frac{240+10.8}{128}) \approx \exp(-0.96) \approx 0.38.$$

Параметризованный фильтр для центрального пикселя (250):

$$I_{out} = 255 \cdot \left(\frac{250}{255}\right)^{0.38} \approx 255 \cdot 0.98 \approx 250.$$

Градиент ∇I измеряет скорость изменения яркости на изображении. Для его расчета используется оператор Собеля.

Для каждого дорожного знака яркость пикселей:

$$\begin{aligned} &[100, 100, 100] \\ &[200, 200, 200] \\ &[200, 200, 200] \end{aligned}$$

Градиент по Собелю:

$$G_x = 200 - 100 = 100,$$

$$\nabla I = \frac{100}{255} \approx 0.39.$$

Для естественного шума (солнечный блик) яркость пикселей:

$$\begin{aligned} &[240, 242, 244] \\ &[242, 245, 247] \\ &[244, 247, 250] \end{aligned}$$

Градиент по Собелю:

$$G_x = 250 - 240 = 10,$$

$$\nabla I = \frac{10}{255} \approx 0.04.$$

Функция проводимости $c(|\nabla I|)$:

$$c(0.39) = \frac{1}{1 + (\frac{0.39}{0.1})^2} = 0.1,$$

$$c(0.04) = \frac{1}{1 + (\frac{0.04}{0.1})^2} = 0.9.$$

Солнечный блик сглаживается, контуры дорожных знаков остаются четкими.

В слое «Optimization» при $E = 60\%$ энергонезависимый коэффициент $\alpha(E)$ равен 1.0.

Фильтрация ложных срабатываний:

$$Score_{final} = 0.9 \cdot \left(1 - \frac{0.2}{0.5}\right) \cdot 1 = 0.54.$$

Калибровка координат:

$$x_{new} = 100 + 0.1 \cdot (105 - 100) = 100.5.$$

$$\alpha_t = 0.01 \cdot (0.456 \cdot 0.9 \cdot 1 \cdot 1 \cdot 3) = 0.012.$$

Постепенная калибровка помогает модели смещаться в сторону правильных значений, улучшая точность распознавания объектов.

Таким образом, модель с использованием рассеивающего фильтра устраняет солнечный блик, обеспечивая высокую точность распознавания дорожного знака.

На рисунке 3 представлен пример распознавания одного из трех дорожных знаков в прикладной программе до внедрения метода интеграции слоев в архитектуру модели, точность которого составляет $P = 39\%$.

Для тестирования модели использовался смартфон марки Xiaomi HyperOS с техническими характеристиками: операционная система Android (версия 14), процессор MediaTek Helio G91-Ultra, оперативная память 8 Гб, внутренняя память 128 Гб.



Рис. 3 – Пример распознавания до внедрения метода интеграции слоев

Fig. 3 – Example of recognition before implementation of the layer integration method

На рисунке 4 представлен пример распознавания одного из трех дорожных знаков в мобильном приложении «Ассистент водителя» после внедрения метода интеграции слоев в архитектуру модели, точность которого составляет $P = 66,99\%$.



Рис. 4 – Пример распознавания после внедрения метода интеграции слоев

Fig. 4 – Example of recognition after implementation of the layer integration method

Аналогично проводятся численные расчеты параметров слоев «Adaptability» и «Optimization» для других дорожных сцен. Численный метод интеграции в архитектуру слоев реализован в прикладной программе, окно которого представлено на рисунке 5 [16].

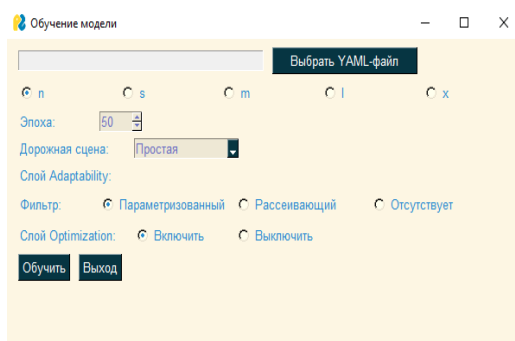


Рис. 5 – Окно «Обучение модели»

Fig. 5 – Model Training Window

Во вкладке «Обучение модели» расположены следующие разделы [16]: «Выбор YAML-файла» (загрузка файла в формате «yaml»); «Выбор модификации» (проводится выбор из 5 модификаций модели: n, s, m, l, x); «Эпоха» (задается количество циклов обучения модели); «Дорожная сцена» (проводится выбор сцен на основании выборки данных: простая, сложная, комбинированная); «Фильтр» (проводится выбор фильтра: параметризованный, рассеивающий); «Слой Optimization» (при включении уменьшается количество эпох обучения, при выключении - увеличивается); «Обучить» (обеспечивает процесс обучения модели); «Выход» (возвращает на главное окно прикладной программы).

Параметризованный фильтр используется для адаптации модели к дождю, снегу, а рассеивающий фильтр - к солнечному блику и туману.

После обучения модели YOLOv8n и YOLOv8n-AE проведен расчет их энергоэффективности. Для расчета энергоэффективности разработанной модели YOLOv8n-AE использовано мобильное устройство ($W = 4$ Вт), а для модели YOLOv8n – портативный компьютер ($W = 25$ Вт).

$$t = 742 \text{ мс} = 0,742 \text{ с};$$

$$FPS = \frac{1}{0,742} \approx 1,35;$$

$$\Theta = \frac{1,35}{4} \approx 0,34 \frac{FPS}{\text{Вт}}.$$

Сравнительная характеристика рассчитанных параметров моделей представлена в таблице 1.

Таблица 1 – Результаты значения энергоэффективности моделей

Table 1 – Energy efficiency ratings of models

Модель	F1-score	Размер модели, МБ	Время обработки кадра, мс	FPS, кадр/с	Энергоэффективность (Θ), $\frac{FPS}{\text{Вт}}$
Стандартная модель					
YOLOv8n	0,42	6,11	854	833	33,32
Адаптивная энергоэффективная модель					
YOLOv8n-AE	0,499	5,95	742	322,58	0,34

Как видно из таблицы 1, разработанная адаптивная энергоэффективная модель YOLOv8n-AE расходует меньше энергии в мобильном устройстве, чем при использовании модели YOLOv8n в портативном компьютере.

Закключение

Таким образом, разработан численный метод интеграции слоев в архитектуру адаптивной энергоэффективной модели для решения задачи распознавания объектов в дорожных сценах. Данный метод позволяет повысить точность распознавания объектов дорожной инфраструктуры (дорожных знаков, дорожных разметок, светофоров) в условиях естественных шумов (дождь, солнечный блик, туман,

снег, сумерки) и энергоэффективность по сравнению со стандартными моделями YOLOv8. Разработанные модели используются в прикладной программе и мобильном приложении «Ассистент водителя» для распознавания перечисленных объектов.

Литература

1. Навой Д. В., Капский Д.В., Филиппова Н.А., Пугачев И.Н., Системный анализ и прикладная информатика, 1, 26-36 (2024).
2. Ключникова В.Д., Федоркина И.А., Баранникова И.В., Тенденции развития науки и образования, 107-8, 16-19 (2024).
3. Артамонов А.А., Дашкин Д.Ш., Пекло Е.А., Молодежная школа-семинар по проблемам управления в технических системах имени А.А. Вавилова, 1, 16-19 (2024).
4. Михалев О.Н., Янюшкин А.С., Мехатроника, автоматика и робототехника, 13, 65-69 (2024).
5. Шепелев С.В., Бабин А.И., Международный журнал прикладных и фундаментальных исследований, 1, 28-32 (2024).
6. Хусаинов Р.М., Вестник технологического университета, 28, 1, 132-136 (2025).
7. Hussain M., Machines, 11, 677 (2023).
8. Пармузин И.И., Электронный журнал: наука, техника и образование, 1 (45), 148-152 (2024).
9. Шульга Т.Э., Солопекин Д.А., Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика, 2, 85-95 (2024).
10. Бурнаева Е.М., Рудаков А.И., Управление образованием: теория и практика, 7 (65), 103-107 (2023).
11. Пипия Л.К., Дорогокупец В.С., Наука за рубежом, 116, 1-55 (2023).
12. Чулков Д.А., Мартышкин А.И., Бершадская Е.Г., Современные информационные технологии, 38 (38), 32-34 (2023).
13. Ковалев В.В., Масленникова Е.А., Филатов Д.М., Федоркова А.О., Международная конференция по мягким вычислениям и измерениям, 1, 246-249 (2024).
14. Макаров П.М., Русскин В.Д., Пашенцев В.Д., Сафонова Т.В., Мокряк А.В., Информационные технологии и системы: управление, экономика, транспорт, право, 4 (48), 4-11 (2023).
15. Дашкин Д.Ш., Артамонов А.А., Пекло Е.А., Сафонова Т.В., Мокряк А.В., Молодежная школа-семинар по

проблемам управления в технических системах имени А.А. Вавилова, 1, 29-32 (2024).

16. Хусаинов Р.М., Вестник технологического университета, 27, 12, 153-157 (2024).

References

1. Navoy D. V., Kapsky D. V., Filippova N. A., Pugachev I. N., System Analysis and Applied Informatics, 1, 26-36 (2024).
2. Klyuchnikova V.D., Fedorkina I.A., Barannikova I.V., Trends in the Development of Science and Education, 107-8, 16-19 (2024).
3. Artamonov A.A., Dashkin D.Sh., Peklo E.A., Youth School-Seminar on Management Problems in Technical Systems named after A.A. Vavilov, 1, 16-19 (2024).
4. Mikhalev O.N., Yanyushkin A.S., Mechatronics, Automation and Robotics, 13, 65-69 (2024).
5. Shepelev S.V., Babin A.I., International Journal of Applied and Fundamental Research, 1, 28-32 (2024).
6. Khusainov R.M., Herald of Technological University, 28, 1, 132-136 (2025).
7. Hussain M., Machines, 11, 677 (2023).
8. Parmuzin I.I., Electronic Journal: Science, Technology and Education, 1 (45), 148-152 (2024).
9. Shulga T.E., Solopekin D.A., Bulletin of Astrakhan State Technical University. Series: Management, Computing Technology and Informatics, 2, 85-95 (2024).
10. Burnayeva E.M., Rudakov A.I., Education Management: Theory and Practice, 7 (65), 103-107 (2023).
11. Pipiya L.K., Dorogokupets V.S., Science Abroad, 116, 1-55 (2023).
12. Chulkov D.A., Martyshkin A.I., Bershadskaya E.G., Modern Information Technologies, 38 (38), 32-34 (2023).
13. Kovalev V.V., Maslennikova E.A., Filatov D.M., Fedorkova A.O., International Conference on Soft Computing and Measurements, 1, 246-249 (2024).
14. Makarov P.M., Russkin V.D., Pashentsev V.D., Safonova T.V., Mokryak A.V., Information Technologies and Systems: Management, Economics, Transport, Law, 4 (48), 4-11 (2023).
15. Dashkin D.Sh., Artamonov A.A., Peklo E.A., Safonova T.V., Mokryak A.V., Youth School-Seminar on Management Problems in Technical Systems named after A.A. Vavilov, 1, 29-32 (2024).
16. Khusainov R.M., Herald of Technological University, 27, 12, 153-157 (2024).

© Р. М. Хусаинов – аспирант кафедры Систем информационной безопасности, Казанский национальный исследовательский технический университет им. А.Н. Туполева, Казань, Россия, rumil_husainov98@mail.ru.

© R. M. Khusainov – PhD-student of Information Security Systems Department, Kazan National Research Technical University named after A.N. Tupolev, Kazan, Russia, rumil_husainov98@mail.ru.

Дата поступления рукописи в редакцию – 15.05.25.

Дата принятия рукописи в печать – 27.05.25.