

**Данильченко Владислав Иванович** – Южный федеральный университет; e-mail: vdaniilchenko@sfedu.ru; г. Таганрог, Россия, тел.: 88634371651; кафедра систем автоматизированного проектирования им. В.М. Курейчика; доцент.

**Бова Виктория Викторовна** – Южный федеральный университет; e-mail: vbova@sfedu.ru; г. Таганрог, Россия, тел.: 88634371651; кафедра систем автоматизированного проектирования им. В.М. Курейчика; доцент.

**Семенова Марина Максимовна** – Южный федеральный университет; e-mail: bogomolova@sfedu.ru; г. Таганрог, Россия, тел.: 88634371651; кафедра систем автоматизированного проектирования им. В.М. Курейчика; аспирант.

**Игнатьева Сабина Валерьевна** – Южный федеральный университет; e-mail: signateva@sfedu.ru; г. Таганрог, Россия, тел.: 88634371651; кафедра систем автоматизированного проектирования им. В.М. Курейчика; студент.

**Шайлиев Марат Баталбиевич** – Южный федеральный университет; e-mail: shayliev@sfedu.ru; г. Таганрог, Россия, тел.: 88634371651; кафедра систем автоматизированного проектирования им. В.М. Курейчика; аспирант.

**Daniilchenko Vladislav Ivanovich** – Southern Federal University; e-mail: vdaniilchenko@sfedu.ru; Taganrog, Russia; phone: +78634371651; V.M. Kureichik Computer-Aided Design Systems Department; associate professor.

**Bova Victoria Viktorovna** – Southern Federal University; e-mail: vbova@sfedu.ru; Taganrog, Russia; phone: +78634371651; V.M. Kureichik Computer-Aided Design Systems Department; associate professor.

**Semenova Marina Maksimovna** – Southern Federal University; e-mail: bogomolova@sfedu.ru; Taganrog, Russia; phone: +78634371651; V.M. Kureichik Computer-Aided Design Systems Department; post-graduate student.

**Ignateva Sabina Valerievna** – Southern Federal University; e-mail: signateva@sfedu.ru; Taganrog, Russia; phone: +788634371651; V.M. Kureichik Department of Computer-Aided Design Systems; student.

**Shayliev Marat Batalbievich** – Southern Federal University; Email: shayliev@sfedu.ru; Taganrog, Russia; phone: +7 88634371651; V.M. Kureichik Department of Computer-Aided Design Systems; postgraduate student.

УДК 004.93

DOI 10.18522/2311-3103-2026-1-20-33

**А.Р. Недвигин, Р.М. Синецкий****АЛГОРИТМИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ОЦЕНКИ ДОСТУПНОСТИ ТОВАРОВ  
В РАСПРЕДЕЛЕННЫХ СИСТЕМАХ ХРАНЕНИЯ НА ОСНОВЕ МЕТОДОВ  
КОМПЬЮТЕРНОГО ЗРЕНИЯ**

*Представлена формализация задачи автоматизированного мониторинга наличия товаров на полках торговых объектов, и соответствия их расположения установленной планогамме с применением методов компьютерного зрения и машинного обучения. Целью исследования является разработка алгоритмического обеспечения для автоматической оценки уровня доступности товаров в распределенных системах хранения с применением методов компьютерного зрения, что позволит разрешить проблему обеспечения оптимального и необходимого ассортимента за счет контроля размещения товаров и поддержки принятия управленческих решений. Предложен технологический процесс обработки визуальных данных, включающий этапы нормализации, сегментации, локализации и классификации объектов с применением сверточных нейронных сетей, в частности YOLO и U-Net. Разработан интегральный показатель доступности, учитывающий физическую, визуальную и информационную составляющие. Сформулирована оптимизационная задача повышения уровня доступности и реализован механизм адаптивного дообучения нейронных сетей, обеспечивающий рост точности распознавания и сегментации изображений и качества аналитических рекомендаций. Предложен алгоритм повышения доступности для системы поддержки принятия решения, основанный на построении маршрута мерчендайзера, который учитывает приоритетность товаров и минимизирует временные затраты. Задача построения*

маршрута мерчандайзера сведена к обобщенной TSP-задаче с весами приоритетов. Предложены и описаны методы оценки и повышения уровня доступности товаров. На основе предложенных методов и алгоритмов разработан программный комплекс для оценки и повышения показателя доступности товаров. Результаты исследования подтверждают эффективность предложенных решений - средняя точность распознавания составила 95,8%, интегральный показатель доступности  $A = 0.93$ . Практическая значимость работы заключается в обеспечении алгоритмической основы для создания интеллектуальных систем мониторинга доступности товарных запасов, способствующих повышению эффективности управления бизнес-процессами.

Мониторинг товаров; системы хранения; компьютерное зрение; сверточные нейронные сети; машинное обучение.

A.R. Nedvigin, R.M. Sinetsky

### METHODOLOGICAL SUPPORT FOR ASSESSING THE AVAILABILITY OF GOODS IN DISTRIBUTED STORAGE BASED ON COMPUTER VISION METHODS

*This paper presents a formalization of the problem of automated monitoring of product availability on retail shelves and compliance with the prescribed planogram, leveraging computer vision and machine learning techniques. The aim of this research is to develop algorithmic solutions for the automatic assessment of product availability in distributed retail environments using computer vision methods, thereby addressing the challenge of maintaining optimal and necessary product assortments through continuous shelf monitoring and supporting data-driven managerial decision-making. A technological pipeline for visual data processing is proposed, comprising the stages of image normalization, segmentation, object localization, and classification, implemented with convolutional neural networks—specifically YOLO and U-Net architectures. An integrated product availability metric is introduced, which jointly accounts for physical, visual, and informational dimensions of availability. An optimization problem aimed at improving overall availability is formulated, and an adaptive neural network fine-tuning mechanism is implemented to enhance the accuracy of image recognition and segmentation, as well as the quality of analytical recommendations. Furthermore, an availability-improvement algorithm is proposed for a decision support system, based on the construction of an optimized merchandiser routing plan that prioritizes products and minimizes time expenditures. This routing problem is reduced to a generalized Traveling Salesman Problem (TSP) with priority-based weights. Methods for evaluating and enhancing product availability are proposed and described in detail. Based on the developed approaches and algorithms, a software system for monitoring and improving product availability has been implemented. Experimental results confirm the effectiveness of the proposed solutions: the average recognition accuracy reached 95.8%, and the integrated availability score achieved  $A = 0.93$ . The practical significance of this work lies in establishing an algorithmic foundation for intelligent shelf-monitoring systems that enable more efficient management of retail operations and inventory processes.*

*Product monitoring; storage systems; computer vision; convolutional neural networks; machine learning.*

**Введение.** Актуальность исследования обусловлена необходимостью повышения эффективности управления товарными запасами и обеспечением высокой доступности товаров в условиях динамичного рынка розничной торговли. Несмотря на развитие технологий компьютерного зрения и аналитических систем, остается нерешенной проблема формализованной количественной оценки уровня доступности и разработки методов ее оптимизации. Задачу автоматизации мониторинга уровня доступности товаров можно рассматривать применительно к сфере розничной торговли как комплексную проблему обеспечения оптимального ассортимента и улучшения покупательского опыта за счет контроля размещения товара на полках и обеспечения принятия управленческих решения в режиме реального времени. Ее можно решить при помощи алгоритмов компьютерного зрения и машинного обучения, объединенных в информационно-аналитической системе, выполняющей следующие функции: обнаружение товаров на изображениях, классификацию по категориям, сверку фактического размещения с планограммой, генерацию рекомендаций, формирование аналитических отчетов.

Такая система может быть создана с применением методов компьютерного зрения (*Computer Vision, CV*) на основе глубокого обучения (*Deep Learning, DL*) [1]. *DL* представляет собой подраздел машинного обучения (*Machine Learning, ML*), которое, в свою

очередь, является областью искусственного интеллекта (рис. 1). *ML* улучшает производительность алгоритмов на заданных задачах путем анализа данных и выявления закономерностей без прямого программирования конкретных решений. *DL* автоматически извлекает высокоуровневые признаки из сырых данных, что делает его эффективным для решения сложных задач, например в области распознавания образов. Одним из наиболее успешных решений в контексте обработки визуальной информации являются сверточные нейронные сети (*Convolutional Neural Networks, CNN*). Это подкласс глубоких нейронных сетей, разработанных для эффективной обработки изображений, предоставляющий переход от низкоуровневых признаков (контур или цвет) к высокому уровню абстракции – объекту в целом.

*CV* фокусируется на анализе и интерпретации содержимого изображений с использованием *DL* (в частности, *CNN*) для обнаружения объектов, сегментации и классификации. Иерархию и взаимосвязь концепций можно описать так: *ML* – общая парадигма системы, *DL* – специализированный инструмент *ML* для работы с неструктурированными данными (изображения, текст, звук), *CV* – прикладная область, в которой *DL* используется для решения задач анализа визуальной информации.

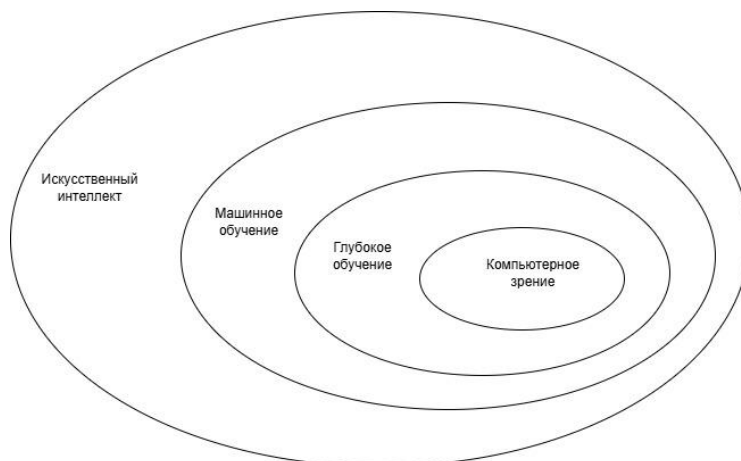


Рис. 1. Схема связи между глубоким обучением и компьютерным зрением

**Обзор современных научных исследований** [2–5] показывает востребованность и актуальность проводимых работ в области автоматизации мониторинга состояния торгового пространства в целом и автоматизации задач оценки доступности товара, в частности. При этом, остается недостаточно изученной и освещенной комплексная задача оптимизации, направленной на повышение уровня доступности товаров. В связи с чем необходимо разработать методику, как совокупность методов и алгоритмов, а также способов их применения для оценки уровня доступности объектов в распределенных системах с применением методов компьютерного зрения, а также сформулировать оптимизационную задачу повышения уровня доступности, предоставив основу для реализации систем для автоматизации решения задач мониторинга состояния товарной представленности и соответствия установленной выкладке.

**Алгоритмы обработки визуальных данных.** В качестве входных данных принимаются изображения торгового оборудования размером  $n \times m$ , полученные при помощи камеры смартфона:

$$I_{input} = \begin{bmatrix} I_{(0,0)} & & \\ & \ddots & \\ & & I_{(n,m)} \end{bmatrix},$$

где  $n$  – число строк,  $m$  – число столбцов.

Изображения могут содержать как непосредственно сами товары на полках, так и элементы фона – стеллажи и прочие объекты. Выходные данные - отчеты о наличии либо отсутствии товаров в установленном количестве, список выявленных отклонений от программы и рекомендации по их устранению. Устанавливаются ограничения системы: точность распознавания отдельных товаров на изображении не менее 95%, оптимизация временных затрат на обработку одного изображения до 1 секунды и возможность масштабирования системы для применения в торговых объектах вне зависимости от их размера.

**Предварительная обработка изображений.** Рассмотрим этапы предобработки:

♦ Нормализация изображений, установка стандартизированного размера  $640 \times 640$  пикселей для унификации входящих данных, чтобы уменьшить вычислительные затраты и облегчить работу модели:

$$I_{normalized}(x, y) = \frac{I_{input}(x, y) - \mu}{\sigma}, \quad x = 0, \dots, m - 1, \quad y = 0, \dots, n - 1,$$

где  $I_{input}$  – исходное изображение размером  $m \times n$  ( $m = 640$ ,  $n = 640$ );  $\mu$  – среднее значение пикселей всего изображения  $I_{input}$ ,  $\sigma$  – стандартное отклонение значений пикселей изображения  $I_{input}$ .

♦ Устранение шумов с применением фильтров (Гаусса и медианного) для удаления артефактов с изображений [6].

Медианный фильтр является локальным оператором, в котором значения пикселя с координатами  $(x, y)$  отфильтрованного изображения  $I_{med}$  вычисляется как медиана значений пикселей исходного изображения в области размером  $K \times K$  ( $K = 2k + 1$ ,  $k \geq 0$ ) вокруг пикселя исходного изображения  $I_{normalized}$  с центром в точке  $(x, y)$ :

$$I_{med}(x, y) = \text{median}(\{I_{normalized}(x + i, y + j)\}), \quad i = -k, \dots, k, \quad j = -k, \dots, k.$$

Фильтр Гаусса реализуется как локальный оператор свертки, ядро которого размером  $K = 2k + 1$ ,  $k \geq 0$  вычисляется как двумерная функция Гаусса:

$$G(i, j) = \frac{1}{2\pi s^2} \exp\left(-\frac{i^2 + j^2}{2s^2}\right), \quad i = -k, \dots, k, \quad j = -k, \dots, k,$$

где  $s$  – параметр фильтрации, задающий интенсивность сглаживания изображения.

Значение пикселя с координатами  $(x, y)$  отфильтрованного изображения  $I_{blur}$  определяется как свертка с исходным изображением  $I_{med}$ :

$$I_{blur}(x, y) = \sum_{i, j = -k}^{+k} I_{med}(x + i, y + j) \cdot G(i, j), \quad x = 0, \dots, m - 1, \quad y = 0, \dots, n - 1.$$

♦ Усиление контрастности в целях улучшения различимости объектов на изображении с использованием нелинейной коррекции изображения методом гистограммной эквализации с приведением гистограммы распределения значений пикселей изображения к нормальному закону [6].

♦ Аугментация данных (повороты, сдвиги, изменения яркости) для увеличения разнообразия обучающих данных и повышения робастности модели и предотвращения возможности переобучения [7].

**Сегментация объектов.** Методы сегментации могут включать классическую сегментацию на основе пороговых значений, либо глубокие нейронные сети для сегментации, например, *U-Net* [8], которые позволяют выделить более сложные структуры. На этом этапе выполняется выделение областей интереса, которые соответствуют полкам либо секциям торгового зала с помощью *CNN*. Цель – ограничить область поиска товаров на изображении, что позволяет значительно ускорить последующую обработку и улучшить производительность системы.

**Локализация объектов.** Для обнаружения и идентификации объектов на изображениях применяется сверточная нейронная сеть *YOLO (You Only Look Once)* [9], которая анализирует изображение целиком и предсказывает классы и ограничивающие рамки

объектов за один проход. *YOLOv9* имеет модульную структуру, состоящую из слоев свертки, блока *Neck* (для обработки и объединения признаков) и блока *Head* (генерирует окончательные предсказания классов и координат объектов). Также предусмотрены слои обработки данных и слой постобработки для удаления дублирующих предсказаний. Математически алгоритм *YOLO* сводится к задаче оптимизации координат объектов и вероятностей их классов:

$$p(x, y) = \sigma_{act}(\theta_i * x + b_i),$$

где  $x, y$  – координаты центра объекта на изображении,  $\theta_i, b$  – веса и смещения для каждого объекта в классе,  $\sigma_{act}$  – функция активации для предсказания вероятности присутствия объектов.

**Процесс предсказания ограничительных рамок в алгоритме YOLO.** Процесс предсказания ограничительных рамок (*bounding box*) является одним из ключевых этапов работы алгоритма [10]. *YOLO* разбивает изображение на равномерную сетку размером  $S \times S$ . Каждая ячейка предсказывает рамки с параметрами:

- ◆ Координаты центра рамки ( $x, y$ ) – нормализованные значения относительно границ ячейки.
- ◆ Размеры рамки ( $w, h$ ) – ширина и высота, выраженные в долях от размеров исходного изображения.
- ◆ Вероятность наличия объекта внутри рамки ( $P_{obj}$ ) – оценка достоверности предсказания.
- ◆ Вероятности класса ( $P(C_1|obj), P(C_2|obj), \dots$ ) – условные вероятности принадлежности объекта к  $C$  – классу, где  $C$  – идентификатор товара.

Для повышения точности предсказания рамок объектов используются якорные рамки (*anchor box*) – заданные эталонные формы, которые адаптируются к типичным пропорциям объектов в выборке (рис. 2). Координаты рамки вычисляются следующим образом:

$$\begin{cases} b_x = \sigma(t_x) + c_x \\ b_y = \sigma(t_y) + c_y \\ b_w = p_w e^{t_w} \\ b_h = p_h e^{t_h} \end{cases},$$

где  $(c_x, c_y)$  – координаты верхнего левого угла ячейки,  $(p_w, p_h)$  – размеры *anchor box*,  $t_x, t_y, t_w, t_h$  – выходные данные нейросети,  $\sigma$  – сигмоидальная функция, ограничивающая смещение центра рамки пределами ячейки.

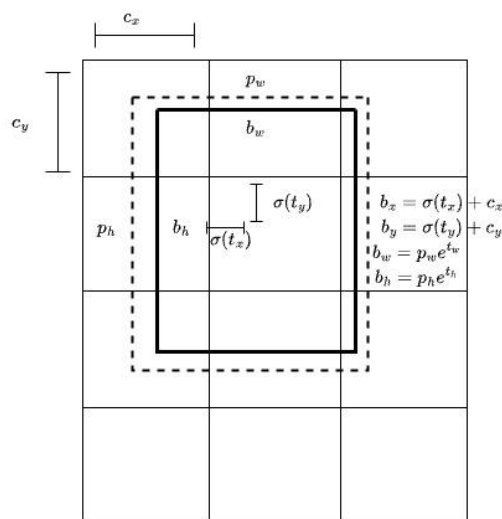


Рис. 2. Предсказание местоположения с использованием *anchor boxes*

Кроме того, для каждой ячейки рассчитываются вероятности  $P_{obj}$ :

$$P_{obj} = C(obj) * IoU(Pred, Truth).$$

Для обучения модели используется составная функция потерь, которая включает компоненты, отвечающие за точность предсказания рамок. Основной метрикой является среднеквадратичная ошибка ( $MSE$ ) между предсказанными и истинными значениями ( $x, y, w, h$ ).

$$\lambda_{MSE} = \sum_{i=1}^N [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 + (w_i - \hat{w}_i)^2 + (h_i - \hat{h}_i)^2],$$

где  $N$  – количество объектов на изображении.

Для фильтрации дублирующих рамок применяется алгоритм *Non-Maximum Suppression (NMS)* [11]. Происходит отсев рамок с  $P_{obj} < \theta$ , где  $\theta$  – порог, подобранный эмпирическим путем. Для каждой вычисляется  $IoU$  [12], и если показатель перекрывающихся рамок превышает заданный порог, то рамка с более низкой оценкой удаляется.

**Классификация товаров.** Классификация товаров выполняется с помощью *CNN*, обученной на заранее размеченных данных и относится к задаче многоклассовой классификации [13], где товар относится к одному из классов, например, категории или бренду. Этапы классификации можно описать так:

- ◆ Извлечение признаков при помощи сверточных слоев, выполняющими фильтрацию изображения для выделения ключевых признаков: контуров, текстур и форм.
- ◆ Классификация с использованием полносвязных слоев, которые вычисляют вероятность принадлежности товара к одному из конкретных классов. Математически процесс классификации можно представить следующим образом:

$$y = f(Wx + b),$$

где  $W$  – веса, обученные в процессе тренировки сети,  $b$  – смещение, а  $x$  – это извлеченные признаки изображения товара,  $f$  – функция активации, в частности, *softmax* для многоклассовой классификации.

**Обнаружение пустых мест.** Выявление пустых участков на полках осуществляется на основе пространственного анализа. Используются морфологические операции эрозии и дилатации, позволяющие выделить области, не содержащие данных [14]:

Эрозию можно представить как операцию уменьшения объекта на изображении:

$$E(x, y) = \min(I(x, y) \oplus S),$$

где  $\oplus$  – операция морфологической свертки, а  $S$  – структурный элемент

Дилатацию же понимают как операцию, которая расширяет объекты:

$$D(x, y) = \min(I(x, y) \odot S),$$

где  $\odot$  – операция морфологической дилатации.

Дополнительно проводится анализ структур для разделения фона и объектов, что позволяет локализовать незаполненные участки [15].

**Постобработка и фильтрация.** После обнаружения товаров важно провести постобработку: удаление ложных срабатываний при помощи фильтров и удаление артефактов (пересекающихся объектов). Для разделения соприкасающихся товаров и устранения искажений из-за частичного перекрытия применяются методы глубокого обучения для сегментации, в частности, использование модели *U-Net* для точного выделения контуров объектов.

**Методы сверки фактического размещения с планограммой.** Планограмма – графическое отображение расположения товаров на полках, преобразуемое в цифровой формат. Рассмотрим три возможных представления: матрица, таблица и геометрическое представление [16].

Матричное представление разбивает полку на сетку, где каждый элемент сетки соответствует определенному месту на полке с данными о товаре.

$$P = \begin{bmatrix} P_{11} & P_{12} & \dots & P_{1n} \\ P_{21} & P_{22} & \dots & P_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ P_{m1} & P_{m2} & \dots & P_{mn} \end{bmatrix},$$

где  $P_{ij}$  – это данные о товаре  $j$ -го типа на  $i$ -й полке.

Преимущества: точное позиционирование и простота анализа. Из недостатков – большие затраты по памяти при увеличении множества товаров.

Табличное представление использует базы данных, присваивая товарам координаты и параметры:

$$T = \{(ID, Category, Quantity, x, y)\},$$

где  $ID$  – это уникальный идентификатор товара,  $Category$  – категория товара,  $Quantity$  – количество товара, которое должно быть на полке,  $x, y$  – координаты товара на полке в двумерной системе.

Табличное представление обладает большей гибкостью и позволяет хранить разнобразную информацию о товаре и обладает возможностями для масштабирования. Недостаток: нужно явно указывать координаты для каждого товара, что усложняет сравнение с фактическим расположением.

Геометрическое отображает планограмму как набор прямоугольных областей, где каждый объект имеет определенные координаты и размеры:

$$R = \{(x_{min}, y_{min}, x_{max}, y_{max}, ID)\},$$

где  $(x_{min}, y_{min})$  – координаты верхнего левого угла прямоугольника,  $(x_{max}, y_{max})$  – координаты нижнего правого угла прямоугольника,  $ID$  – уникальный идентификатор товара, размещенного в этой области.

Геометрическое представление является наглядным и легко интерпретируемым. Для сверки расположения товаров с планограммой удобно использовать  $IoU$  (*Intersection over Union*). При этом требуется точное указание размеров и положения каждого товара, что увеличивает трудоемкость при работе с большими объемами данных.

**Сопоставление координат и вычисление  $IoU$  для каждого товара.** Для сверки фактического размещения с планограммой используется метрика  $IoU$ . Она определяет степень перекрытия между фактическими координатами товаров с предполагаемыми согласно планограмме:

$$IoU = \frac{S_{intersection}}{S_{union}},$$

где  $S_{intersection}$  – это площадь, где пересекаются фактическое и предполагаемое расположение товара, а  $S_{union}$  – это площадь, занимаемая обеими объединенными областями.

Определяются случаи, в которых  $IoU$  меньше установленного порога (в конкретном случае 0,8). В таком случае фиксируется отклонение, так как считается, что товар расположен вне установленной планограммы, либо отсутствует. Схема процесса сверки размещения товара с планограммой представлена на рис. 3.

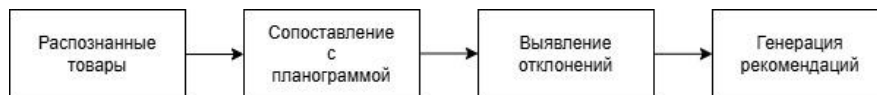


Рис. 3. Схема процесса сверки фактического размещения товара с планограммой

**Методы оценки уровня доступности объектов.** В общем виде цель задачи по оценке уровня доступности товаров на полках розничных магазинов можно сформулировать как определение степени фактического наличия товаров их плановому состоянию, заданному планограммой. Система оценки доступности опирается на концепцию обратной связи, при которой измеренные параметры торгового пространства сравниваются с целевыми значениями и формируют сигнал ошибки, используемый для принятия управленческих решений.

Для оценки уровня доступности введем интегральный показатель  $A$ , учитывающий три составляющих:

1. Физическая доступность (наличие товара на полке).

$$A_f = \frac{N_{present}}{N_{total}},$$

где  $N_{present}$  – количество позиций, фактически присутствующих на полке,  $N_{total}$  – общее количество позиций, подлежащих проверке.

2. Визуальная доступность (доля видимой площади упаковок относительно эталонной)

$$A_v = 1 - \frac{S_{empty}}{S_{total}},$$

где  $S_{empty}$  – площадь пустого пространства, где упаковки отсутствуют, но должны быть по нормативу,  $S_{total}$  – общая площадь для размещения товара по плануграмме.

3. Информационная доступность (соответствие фактического размещения плануграмме).

$$A_i = \frac{N_{correct}}{N_{checked}},$$

где  $N_{correct}$  – количество проверенных позиций, корректно размещенных в соответствии с плануграммой,  $N_{checked}$  – общее количество проверенных позиций, для которых проводилось сравнение с плануграммой.

Интегральный показатель определяется как:

$$A = \omega_1 * A_f + \omega_2 * A_v + \omega_3 * A_i,$$

где  $\omega_1 + \omega_2 + \omega_3 = 1$  – весовые коэффициенты, определяемые методом экспертных оценок или, иначе, методом аналитической иерархии - *AHP* (*Analytic Hierarchy Process*) [17].

При полученном значении  $A$  ниже заданного порога (определяется эмпирически, например 0,8) потребуются корректирующие воздействия для повышения доступности товаров.

**Генерация рекомендаций и алгоритм поддержки принятия решений.** На основе результатов оценки уровня доступности генерируется список действий для устранения выявленных отклонений, например переместить товар на полку согласно плануграмме или добавить недостающий товар для заполнения полки. Но так список рекомендаций будет сформирован без достаточной гибкости, необходимой для систем принятия решений, так как не учитывает приоритеты пополнения товаров. Выделим такие факторы, как товары первой необходимости (индикаторы), влияющие на выбор покупателя, акционные товары, наличие которых определяется договорными обязательствами, высокомаржинальные товары (продукция собственной торговой марки). Соответственно, для уточненных рекомендаций необходимо внести в алгоритм поправочные коэффициенты на основе статистики продаж. В рамках данной работы предложено упрощение модели с использованием базовых показателей:  $s$  – средняя оборачиваемость товара за день (шт. /день),  $p$  – цена товара (руб.),  $u$  – срочность ( $0 \dots 100$ ), где 100 – максимальная срочность;

Для каждого обнаруженного класса  $i$  введем коэффициент приоритета  $P_i$ :

$$P_i = w_s * \frac{s_i}{\max_j s_j} + w_p * \frac{p_i}{\max_j p_j} + w_u * \frac{u_i}{100},$$

где  $w_s, w_p, w_u$  – весовые коэффициенты (сумма которых равна 1), нулевое значение является собой отсутствие отклонений, а единица – максимальное отклонение, требующее приоритетного устранения.

Для определения весовых коэффициентов используется метод *AHP*. Пример расчета: позиция  $A$  имеет  $s_A = 0,8$ , что эквивалентно 80% от максимального объема продаж,  $p_A = 0,5$  (половина от максимальной цены),  $u_A = 1$  (высокая срочность). При  $w_s = 0,3$ ,  $w_p = 0,2$ ,  $w_u = 0,5$  мы получим  $P_A = 0,84$ .

**Построение маршрута мерчандайзера.** Пусть мерчандайзер получил список товарных позиций (*stock keeping unit, SKU*)  $\{i_1, i_2, \dots, i_k\}$  с приоритетами  $\{P_1, P_2, \dots, P_k\}$ . Построение маршрута мерчандайзера сводится к задаче коммивояжера с весами приоритетов (*travelling salesman problem, TSP*) [18]. Целью является минимизация общего расстояния и времени (стоимости) пути с учетом приоритетов. Задача формулируется как

обобщенная *TSP* с весами приоритетов, где граф  $G = (V, E)$ , где  $V$  – набор точек, представляющих собой товарные позиции на полке,  $E$  – ребра с весами  $d(v_i, v_j)$ . Оптимальный маршрут  $\pi$  определяется как минимум от обобщенной длины:

$$L(\pi) = \sum_{t=1}^{k-1} d(v_{\pi(t)}, v_{\pi(t+1)}) + \alpha \sum_{t=1}^k (t-1)(1 - P_{\pi(t)}),$$

где первый член – расстояние пути, а второй член – штрафной, который способствует стимулированию посещения узлов с высоким  $P$ .

Используется жадный алгоритм с оценочной функцией:  $Score(j) = \lambda d(v_{current}, v_j) - (1 - \lambda)P_j$ , где  $\lambda \in [0,1]$  – параметр между сокращением расстояния и учетом приоритета. Ниже пример расчета маршрута мерчандайзера. Данные для расчета примера см. в табл. 1.

Таблица 1

**Исходные данные**

SKU	Оборачиваемость $s_i$ (шт./день)	Стоимость $p_i$ (руб)	Срочность $u_i$ (%)	X(м)	Y(м)
Позиция А	120	100	100	20	8
Позиция В	80	150	40	5	15
Позиция С	100	120	60	8	2
Позиция D	50	80	20	3	3
Позиция Е	150	20	100	7	18

- ◆ Оборачиваемость – среднее число проданных единиц товара в день.
- ◆ Стоимость – средняя цена товара.
- ◆ Срочность – показывает отличие наличие товара отличается от заданного в планеграмме. Примем, что А и Е полностью отсутствуют, т.е. имеют отклонение равное 100%, актуальный остаток В составляет 60%, т.е. отклонение 40% и т.д.
- ◆ Координаты X, Y – положение товаров на плане торгового зала (метры от стартовой точки (0,0)). Каждый показатель нормируется к диапазону [0,1] (табл. 2) по максимальному значению в столбце:  $\acute{s}_i = \frac{s_i}{\max(s)}$ ,  $\acute{p}_i = \frac{p_i}{\max(p)}$ ,  $\acute{u}_i = \frac{u_i}{100}$ .

Таблица 2

**Данные после нормирования**

SKU	$\acute{s}_i$	$\acute{p}_i$	$\acute{u}_i$
Позиция А	0.8	0.67	1.0
Позиция В	0.53	1.0	0.4
Позиция С	0.67	0.8	0.6
Позиция D	0.33	0.53	0.2
Позиция Е	1.0	0.13	1.0

Для расчета приоритета  $P_i$  используем весовые коэффициенты  $w_s = 0,3$ ,  $w_p = 0,2$ ,  $w_u = 0,5$ . Таким образом, получаем:  $P_i = w_s \acute{s}_i + w_p \acute{p}_i + w_u \acute{u}_i$

Для построения принятия решений очередности выполнения заданий используем алгоритмическую парадигму жадной эвристики [19]. Пусть  $\lambda = 0,5$ . На каждом шаге среди непосещенных узлов выбирается тот, у которого минимально значение  $Score(j) = \lambda d(current, j) - (1 - \lambda)P_j$ , где  $d$  – евклидово расстояние от текущей позиции. Таким образом, итоговый маршрут выглядит следующим образом: начало маршрута:

(0,0) – стартовая точка, вход в торговый зал. Последовательность отработки заданий:  $D(3,3) \rightarrow C(8,2) \rightarrow A(20,8) \rightarrow B(5,15) \rightarrow E(7,18) \rightarrow (0,0)$ . График рассчитанного маршрута см. на рис. 4.

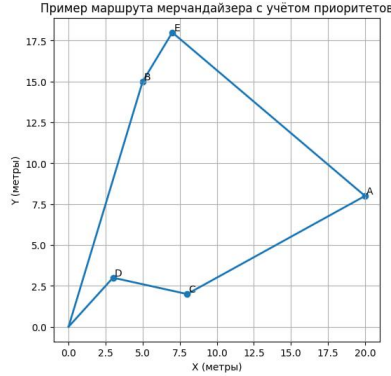


Рис. 4. Пример маршрута мерчендайзера с учетом приоритетов

**Механизм обратной связи для дообучения.** Для минимизации повторяющихся ошибок и сокращения предусмотрим алгоритм цикла обратной связи. В случае, если сотрудник отмечает задание как ошибочное («Отклонить – Ложное срабатывание»), то система отмечает запись соответствующим флагом в *DETECTIONS*, а в таблицу *LOGS* добавляется запись, *action = false\_positive* для соответствующего *detection\_id*. Счетчик ложных срабатываний *false\_positive\_count* в *DETECTIONS* увеличивается при каждом отклонении для класса *i* в рамках одной задачи. При достижении значения «3», формируется запрос на перестройку тренировочной выборки. Ложные примеры добавляются в пакет для дообучения.

Система формирует выборку верно положительных детекций и ложно положительных сигналов в пропорции 4:1. При достижении  $\geq 100$  ложно положительных примеров запускается инкрементальный процесс дообучения без полного переобучения модели. Диаграмма процесса показана на рис. 5.

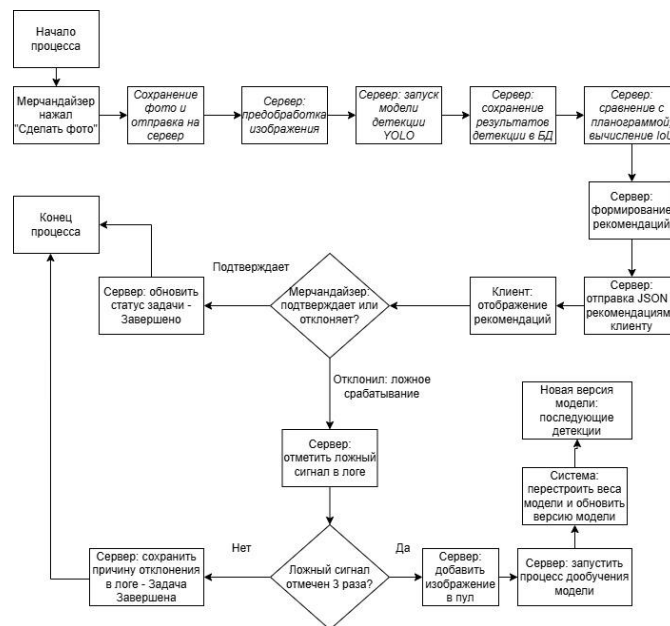


Рис. 5. Диаграмма процесса потока работ до возможного дообучения

**Генерация аналитических отчетов.** Основные типы отчетов включают в себя:

◆ Отчет о доступности товаров, который показывает процент заполнения полок и выделяет пустые места, чтобы оптимизировать запасы.

$$P_{empty} = \frac{S_{empty}}{S_{total}} \times 100,$$

где  $S_{empty}$  – площадь пустых мест на полке,  $S_{total}$  – общая площадь полки.

◆ Отчет о соблюдении планограммы, в котором указываются выявленные отклонения и несоответствия в выкладке товаров.

**Оценка производительности системы.** Для эффективной интеграции системы в торговую среду, необходимо произвести оценку ее производительности. Основные метрики [20]:

◆ Точность распознавания (*Precision*). определяет долю верно классифицированных примеров:

$$P = \frac{TP}{TP+FP},$$

где  $TP$  (*True Positives*) – количество истинно-положительных результатов;  $FP$  (*False Positives*) – количество ложных положительных результатов.

◆ Полнота (*Recall*) - доля корректно классифицированных положительных примеров:

$$P = \frac{TP}{TP+FN},$$

где  $FN$  (*False Negatives*) – количество ложных отрицательных результатов.

◆ Среднее время обработки одного изображения.

◆ Скорость обработки в реальном времени.

**Задача повышения уровня доступности.** Повышение уровня доступности товаров рассматривается как комплексная задача оптимизации, которая включает в себя диагностический, аналитический, оптимизационный и корректирующий этапы.

1. Диагностический этап. Направлен на фиксацию текущего состояния системы и определение факторов, влияющих на снижение доступности. Включает в себя сбор данных, анализ изображения, сверку с эталонной схемой, расчет частных показателей доступности (физической, визуальной, информационной).

2. Аналитический этап. Для каждого найденного класса объектов определяется коэффициент приоритета.

3. Оптимизационный этап. На основе выявленных факторов строится модель оптимизации доступности. Алгоритм определяет оптимальную последовательность действий, минимизирующую совокупные затраты.

4. Корректирующий этап. Рассматривается как формирование управленческих воздействий с созданием перечня корректирующих мероприятий. В их числе перемещение или добавление товаров согласно эталонной планограмме, корректировка приоритетов, а также формирование маршрута обхода с учетом приоритетов с обеспечением минимизации пути и времени и максимизации прироста  $A$ .

5. Этап адаптивного дообучения.

Полный цикл процесса повышения уровня доступности можно представить как замкнутую систему управления (рис. 6).



Рис. 6. Схема процесса повышения уровня доступности объектов

На каждом цикле функция управления  $U(t)$  определяется как:

$$U(t) = f(E_x, P_i, \Delta A_t),$$

где  $P_i$  – приоритет позиции,  $E_x$  – эталонное состояние,  $\Delta A_t = A_{t+1} - A_t$ , что отражает прирост доступности по результатам цикла. При этом, эффективность оценивается сравнением значений  $A_t$  и  $A_{t+1}$ , рост показателя  $A$  свидетельствует о корректности механизмов диагностики и оптимизации.

**Заключение.** Разработаны алгоритмы оценки уровня доступности товаров в распределенных системах хранения с применением методов компьютерного зрения. Предложен интегральный показатель доступности, учитывающий физическую, визуальную и информационную составляющие. Сформулирована оптимизационная задача повышения доступности и реализован механизм адаптивного дообучения, обеспечивающий рост показателя доступности. На основе предложенных методов и алгоритмов разработан программный комплекс [21] для оценки и повышения показателя доступности товаров. Средняя точность распознавания объектов при помощи разработанного программного комплекса составила 95,8%, показатель доступности  $A = 0,93$ . На основании проделанной работы можно утверждать, что предложенная совокупность методов обнаружения идентификации товаров, извлечения признаков, классификации и формирования аналитических отчетов за счет комбинирования методов глубокого и машинного обучения является собой эффективный и надежный способ мониторинга торгового пространства.

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Tiwari Tanya & Tiwari Tanuj & Tiwari Sanjay.* How Artificial Intelligence, Machine Learning and Deep Learning are Radically Different? // *International Journal of Advanced Research in Computer Science and Software Engineering.* – 2018. – 8(2):1. – 10.23956/ijarcsse.v8i2.569.
2. *Mehwish Saqlain, Saddaf Rubab, Malik M. Khan, Nouman Ali, Shahzeb Ali.* Hybrid Approach for Shelf Monitoring and Planogram Compliance (Hyb-SMPC) in Retails Using Deep Learning and Computer Vision // *Mathematical Problems in Engineering.* – 2022. – No. 1. – <https://doi.org/10.1155/2022/4916818>.
3. *Jorge Muñoz, Alonso Sanchez, Guillermo Kemper.* End-to-end solution for automatic beverage stock detection in supermarkets based on image processing and convolutional neural networks // *International Journal of Cognitive Computing in Engineering.* – 2024. – No. 5. – P. 453-474. – <https://doi.org/10.1016/j.ijcce.2024.09.001>.
4. *Wei Yuchen, Tran Son, Xu Shuxiang, Kang Byeong, Springer Matthew.* Deep Learning for Retail Product Recognition: Challenges and Techniques // *Computational Intelligence and Neuroscience.* – 2020. – 8875910. – P. 1-23. – <https://doi.org/10.1155/2020/8875910>.
5. *Gothai E. et al.* Design features of grocery product recognition using deep learning // *Intelligent Automation and Soft Computing.* – 2022. – Vol. 34, No. 2. – P. 1231-1246.
6. *Рейнхард Клетте.* Компьютерное зрение. Теория и алгоритмы: пер. с англ. А.А. Слинкин. – М.: ДМК Пресс, 2019. – 506 с.
7. *Yolo Data Augmentation.* – 2025. – Режим доступа: <https://docs.ultralytics.com/guides/yolo-data-augmentation> (дата обращения: 02.05.2025).
8. *Surono Sugiyarto & Rivaldi Muhammad & Arrova Dewi Deshinta & Irsalinda Nursyiva.* New Approach to Image Segmentation: U-Net Convolutional Network for Multiresolution CT Image Lung Segmentation // *Emerging Science Journal.* – 2023. – 7. – P. 498-506. – 10.28991/ESJ-2023-07-02-014.
9. *YOLOv9 Architecture Explained.* – Режим доступа: <https://stunningvisionai.com/article/yolov9-architecture> (дата обращения: 02.05.2025).
10. *Desai Miral & Mewada Hiren & Pires Ivan & Roy Sparsh.* Evaluating the Performance of the YOLO Object Detection Framework on COCO Dataset and Real-World Scenarios // *Procedia Computer Science.* – 2024. – 251. – P. 157-163. – 10.1016/j.procs.2024.11.096.
11. *Wang Decheng & Chen Xiangning & Yi Hui & Zhao Feng.* Improvement of Non-Maximum Suppression in RGB-D Object Detection // *IEEE Access.* – 2019. – Vol. 7. – 10.1109/ACCESS.2019.2945834.
12. *Бондаренко В.А., Гессен П.А., Павлова В.А., Созинова М.В., Тупиков В.А.* Алгоритм обнаружения объектов для оптико-электронных систем с обучением в реальном времени // *Известия ЮФУ. Технические науки.* – 2021. – № 1 (218).
13. *Бринк Х., Ричардс Д., Феверолф М.* Машинное обучение. – СПб.: Питер, 2017. – 336 с.
14. *Said K & Jambek Asral.* Analysis of Image Processing Using Morphological Erosion and Dilation // *Journal of Physics: Conference Series.* – 2021. – 2071(1):012033. – 10.1088/1742-6596/2071/1/012033.
15. *Santra Bikash & Ghosh Udita & Mukherjee Dipti.* Graph-based Modelling of Superpixels for Automatic Identification of Empty Shelves in Supermarkets // *Pattern Recognition.* – 2022. – 127 (2): 108627. – 10.1016/j.patcog.2022.108627.

16. Liu Song & Li Wanqing & Davis Stephen & Ritz Christian & Tian Hongda. Planogram Compliance Checking Based on Detection of Recurring Patterns // IEEE MultiMedia. – 2016. – Vol. 23. – P. 54-63. – 10.1109/MMUL.2016.19.
17. Баркалов С.А., Карпович М.А., Мусеев С.И. Метод анализа иерархий: подход, основанный на теории латентных переменных // Вестник ЮУрГУ. Серия: Компьютерные технологии, управление, радиоэлектроника. – 2022. – № 2.
18. Matai Rajesh & Singh Surya & Mittal M.L. Traveling Salesman Problem: an Overview of Applications, Formulations, and Solution Approaches. – 2010. – 10.5772/12909.
19. Cal Murat & Ekici Ali. Solving a Modified TSP Problem by a Greedy Heuristic for Cost Minimization // International Journal of Modeling and Optimization. – 2018. – 8. – P. 138-144. – 10.7763/IJMO.2018.V8.638.
20. Недвигин А.Р., Синецкий Р.М. Применение алгоритмов обработки изображений сверточных нейронных сетей в качестве основы для разработки системы поддержки принятия решения // Известия вузов. Северо-Кавказский регион. Серия: Технические науки. – 2025. – № 1.
21. Свидетельство о государственной регистрации программы для ЭВМ № 2025664496 Российская Федерация. Комплексная оценка доступности товаров с использованием технологий компьютерного зрения Ритейл-Диагност. Серверная часть. Версия 1.0 / А.Р. Недвигин, Р.М. Синецкий; Заявл. 15.05.2025; Запат. 04.06.2025.

## REFERENCES

1. Tiwari Tanya & Tiwari Tanuj & Tiwari Sanjay. How Artificial Intelligence, Machine Learning and Deep Learning are Radically Different?, *International Journal of Advanced Research in Computer Science and Software Engineering*, 2018, 8(2):1, 10.23956/ijarcsse.v8i2.569.
2. Mehwish Saqlain, Saddaf Rubab, Malik M. Khan, Nouman Ali, Shahzeb Ali. Hybrid Approach for Shelf Monitoring and Planogram Compliance (Hyb-SMPC) in Retails Using Deep Learning and Computer Vision, *Mathematical Problems in Engineering*, 2022, No. 1. Available at: <https://doi.org/10.1155/2022/4916818>.
3. Jorge Muñoz, Alonso Sanchez, Guillermo Kemper. End-to-end solution for automatic beverage stock detection in supermarkets based on image processing and convolutional neural networks, *International Journal of Cognitive Computing in Engineering*, 2024, No. 5, pp. 453-474. Available at: <https://doi.org/10.1016/j.ijcce.2024.09.001>.
4. Wei Yuchen, Tran Son, Xu Shuxiang, Kang Byeong, Springer Matthew. Deep Learning for Retail Product Recognition: Challenges and Techniques, *Computational Intelligence and Neuroscience*, 2020, 8875910, pp. 1-23. Available at: <https://doi.org/10.1155/2020/8875910>.
5. Gothai E. et al. Design features of grocery product recognition using deep learning, *Intelligent Automation and Soft Computing*, 2022, Vol. 34, No. 2, pp. 1231-1246.
6. Reynkhard Klette. Компьютерное зрение. Теория и алгоритмы [Computer vision. Theory and algorithms]: transl. from Engl. by A.A. Slinkin. Moscow: DMK Press, 2019, 506 p.
7. Yolo Data Augmentation, 2025. Available at: <https://docs.ultralytics.com/guides/yolo-data-augmentation> (accessed 02 May 2025).
8. Surono Sugiyarto & Rivaldi Muhammad & Arrova Dewi Deshinta & Irsalinda Nursyiva. New Approach to Image Segmentation: U-Net Convolutional Network for Multiresolution CT Image Lung Segmentation, *Emerging Science Journal*, 2023, 7, pp. 498-506. 10.28991/ESJ-2023-07-02-014.
9. YOLOv9 Architecture Explained. Available at: <https://stunningvisionai.com/article/yolov9-architecture> (accessed 02 May 2025).
10. Desai Miral & Mewada Hireen & Pires Ivan & Roy Sparsh. Evaluating the Performance of the YOLO Object Detection Framework on COCO Dataset and Real-World Scenarios, *Procedia Computer Science*, 2024, 251, pp. 157-163. 10.1016/j.procs.2024.11.096.
11. Wang Decheng & Chen Xiangning & Yi Hui & Zhao Feng. Improvement of Non-Maximum Suppression in RGB-D Object Detection, *IEEE Access*, 2019, Vol. 7. 10.1109/ACCESS.2019.2945834.
12. Bondarenko V.A., Gessen P.A., Pavlova V.A., Sozinova M.V., Tupikov V.A. Algoritm obnaruzheniya ob"ektov dlya optiko-elektronnykh sistem s obucheniem v real'nom vremeni [Object detection algorithm for optoelectronic systems with online learning], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2021, No. 1 (218).
13. Brink Kh., Richards D., Feverolf M. Mashinnoe obuchenie [Machine learning]. St. Petersburg: Piter, 2017, 336 p.
14. Said K & Jambek Asral. Analysis of Image Processing Using Morphological Erosion and Dilation, *Journal of Physics: Conference Series*, 2021, 2071(1):012033. 10.1088/1742-6596/2071/1/012033.

15. Santra Bikash & Ghosh Udit & Mukherjee Dipti. Graph-based Modelling of Superpixels for Automatic Identification of Empty Shelves in Supermarkets, *Pattern Recognition*, 2022, 127(2): 108627. 10.1016/j.patcog.2022.108627.
16. Liu Song & Li Wanqing & Davis Stephen & Ritz Christian & Tian Hongda. Planogram Compliance Checking Based on Detection of Recurring Patterns, *IEEE MultiMedia*, 2016, Vol. 23, pp. 54-63. 10.1109/MMUL.2016.19.
17. Barkalov S.A., Karpovich M.A., Moiseev S.I. Metod analiza ierarkhiy: podkhod, osnovanny na teorii latentnykh peremennykh [Hierarchy analysis method: an approach based on latent variable theory], *Vestnik YuUrGU. Seriya: Komp'yuternye tekhnologii, upravlenie, radioelektronika* [Bulletin of SUSU. Series: Computer Technologies, Control, Radio Electronics], 2022, No. 2.
18. Matai Rajesh & Singh Surya & Mittal M.L. Traveling Salesman Problem: an Overview of Applications, Formulations, and Solution Approaches, 2010. 10.5772/12909.
19. Cal Murat & Ekici Ali. Solving a Modified TSP Problem by a Greedy Heuristic for Cost Minimization, *International Journal of Modeling and Optimization*, 2018, 8, pp. 138-144. 10.7763/IJMO.2018.V8.638.
20. Nedvigina A.R., Sinetskiy R.M. Primenenie algoritmov obrabotki izobrazheniy svertochnykh neyronnykh setey v kachestve osnovy dlya razrabotki sistemy podderzhki priyatiya resheniya [Application of image processing algorithms of convolutional neural networks as a basis for developing a decision support system], *Izvestiya vuzov. Severo-Kavkazskiy region. Seriya: Tekhnicheskie nauki* [Bulletin of Higher Educational Institutions. North Caucasus Region. Technical Sciences], 2025, No. 1.
21. Nedvigina A.R., Sinetskiy R.M. Svidetel'stvo o gosudarstvennoy registratsii programmy dlya EVM № 2025664496 Rossiyskaya Federatsiya. Kompleksnaya otsenka dostupnosti tovarov s ispol'zovaniem tekhnologii komp'yuternogo zreniya Riteyl-Diagnost. Servernaya chast'. Versiya 1.0 [Certificate of State Registration of a Computer Program No. 2025664496, Russian Federation. Comprehensive Assessment of Product Availability Using Computer Vision Technologies – Retail-Diagnost. Server Side. Version 1.0]; Filed on: 15.05.2025; Registered on: 04.06.2025.

**Недвигин Артем Романович** – Южно-Российский государственный политехнический университет (НПИ) имени М.И. Платова, e-mail: artemnedvigin@mail.ru; г. Новочеркасск, Россия; тел: 89185573541; кафедра «Программное обеспечение вычислительной техники»; аспирант.

**Синецкий Роман Михайлович** – Южно-Российский государственный политехнический университет (НПИ) имени М.И. Платова, e-mail: rmsin@srsru.ru; г. Новочеркасск, Россия; тел: 89289601848; кафедра «Программное обеспечение вычислительной техники»; к.т.н.; доцент.

**Nedvigina Artem Romanovich** – Platov South-Russian State Polytechnic University (NPI); e-mail: artemnedvigin@mail.ru; Novocherkassk, Russia; phone: +79185573541; the Department of Computer Software; graduate student.

**Sinetskiy Roman Mikhailovich** – Platov South-Russian State Polytechnic University (NPI); e-mail: rmsin@srsru.ru; Novocherkassk, Russia; phone: +79289601848; the Department of Computer Software; cand. of eng. sc.; associate professor.

УДК 681.327

DOI 10.18522/2311-3103-2026-1-33-42

**П.О. Никашина**

### **АЛГОРИТМ ОПРЕДЕЛЕНИЯ НЕЧЕТКОГО МНОЖЕСТВА СИЛЬНОЙ СВЯЗНОСТИ ПЕРИОДИЧЕСКОГО НЕЧЕТКОГО ГРАФА**

*Рассматривается метод определения сильной связности периодического нечеткого графа (PFG), который может быть применен для анализа динамических систем с учетом неопределенности и изменений во времени. Вводится понятие нечеткого множества сильной связности, которое позволяет оценить степень достижимости между вершинами графа за определенное количество тактов времени. Основное внимание уделяется разработке алгоритма для нахождения данного множества, что является важным инструментом для анализа связности в сложных системах. Статья начинается с обзора существующих подходов к анализу связности в нечетких графах, подчеркивая необходимость учета временных и нечетких параметров. Основная часть работы посвящена описанию ключевых понятий и определений, связанных с периодическими не-*