Интеллектуальный сервис мультимодального мониторинга области наблюдения

Р. Р. Миннеахметов[1] **[0009-0007-8551-1393]** *[razil0071999@gmail.com]*

[1] *Казанский федеральный университет, Институт информационных технологий и интеллектуальных систем, г. Казань, ул. Кремлевская, 35*

Аннотация. Представлен подход к построению интеллектуального сервиса мультимодального мониторинга области наблюдения с использованием больших нейросетевых моделей. Предложенный метод ориентирован на анализ разнородных данных: видеопотоков, сигналов от сенсоров (температура, влажность) и логов событий, поступающих из наблюдаемой зоны.

В качестве основных инструментов использованы крупные языковые и визуальные модели (LLaMA, MiniCPM-V и др.), развёрнутые с помощью локального фреймворка Ollama, обеспечивающего автономную и безопасную обработку информации. Разработан и протестирован прототип сервиса, способный выявлять критические ситуации, отклонения от нормы и контекстно значимые события в режиме офлайн.

Описана методика формирования сценариев и проведения оценки качества (F1-Score, Precision, Recall) на множестве тестов с различными объектами и ситуациями. Экспериментальные результаты подтверждают применимость мультимодальных моделей для задач мониторинга и демонстрируют их потенциал в построении адаптивных и масштабируемых систем наблюдения.

Ключевые слова: интеллектуальный сервис, мультимодальный мониторинг, Ollama, большие языковые модели, отслеживание активностей, видеоаналитика, искусственный интеллект

Intelligent multimodal monitoring service for the surveillance area

R. Minneakhmetov[1] **[0009-0007-8551-1393]** *[razil0071999@gmail.com]*

[1] *Kazan Federal University, Institute of Information Technologies and Intellectual Systems, Kazan, Kremlevskaya ul. 35*

Abstract. This paper presents an approach to developing an intelligent multimodal monitoring service for surveillance areas using large neural network models. The proposed method focuses on analyzing heterogeneous data sources – video streams, environmental sensor signals (e.g., temperature, humidity), and event logs – within the observed domain.

The system leverages advanced language and vision models (e.g., LLaMA, MiniCPM-V), deployed locally via the Ollama framework, enabling secure and autonomous processing without cloud dependency. A functional prototype has been implemented and tested to detect critical situations, abnormal patterns, and context-specific events offline.

The data processing methodology and experimental evaluation based on predefined scenarios are described. Results demonstrate the effectiveness of multimodal models for activity monitoring and highlight their potential in building adaptive and scalable surveillance systems.

Keywords: intelligent service, multimodal monitoring, Ollama, Large Language Models, activity tracking, video analytics, artificial intelligence

Введение

Современные интеллектуальные системы все чаще применяются для анализа поведенческих и ситуационных паттернов в реальном времени. Одним из перспективных направлений является **мультимодальный мониторинг** [1] — то есть анализ информации, поступающей одновременно из различных источников: видеопотоков, сенсорных сигналов (например, температуры и влажности), текстовых логов и прочих структурированных или неструктурированных данных. Такой подход позволяет получить более полную картину происходящего и повысить достоверность выводов за счёт перекрёстной верификации данных из разных модальностей.

Под **областью наблюдения** в рамках данной работы понимается ограниченное физическое или логическое пространство, в котором осуществляется автоматическое отслеживание активности. Это может быть, например, помещение, коридор, производственный участок или виртуальная зона, контролируемая с помощью видеонаблюдения, сенсоров или журналируемых событий.

В отличие от классических систем безопасности, основанных на сигнатурах и ручной настройке правил, предлагаемый интеллектуальный сервис использует возможности **больших нейросетевых моделей (Large Language Models и Vision-моделей)** для автономного анализа событий, применяющиеся в различных сферах, включая обработку естественного языка, компьютерное зрение, анализ временных рядов [2]. В системах мониторинга их использование позволяет автоматизировать распознавание сложных паттернов поведения и потенциально опасных действий, что ранее требовало значительных вычислительных ресурсов и ручного вмешательства [3]. В отличие от традиционных систем обнаружения инцидентов, основанных на сигнатурах и правилах [4], проактивный мониторинг поведения пользователей нацелен на поиск аномалий, не

выявленных стандартными средствами защиты [5]. Для эффективного выявления подозрительных отклонений активно применяются методы машинного обучения (Machine Learning, ML) [6], способные обнаруживать нетипичную активность на основе исторических данных [5, 7]. Под активностью понимается любое изменение конфигурации любой наблюдаемой зоны, к примеру события в системе журналирования или действия на кадрах видеонаблюдения.

Настоящая работа посвящена разработке прототипа такого сервиса на базе локального фреймворка Ollama [8], обеспечивающего безопасную и автономную работу моделей на персональном устройстве без отправки данных на удаленные серверы. В статье подробно рассматриваются методология обработки мультимодальных данных, выбор моделей, а также результаты их тестирования и сравнительной оценки.

Обзор существующих решений

Для эффективного отслеживания активностей необходимо анализировать разнородные данные: видеопотоки, логи систем, показания носимых сенсоров, а также контент, создаваемый пользователями (User Generated Content, UGC). Предлагаемая методология базируется на использовании предобученных больших нейросетевых моделей, способных обрабатывать эти данные и извлекать из них значимые паттерны [9]. В части компьютерного зрения применяются глубокие сверточные сети и Vision Transformer-модели для распознавания действий на видео и классификации поведения людей в реальном времени [10]. В проведенном анализе последовательностей сенсорных сигналов (ускорение, гироскоп и др.) используются архитектуры на базе LSTM/GRU или Transformer, обученные на больших наборах данных о движениях. Это позволяет моделям выявлять характерные последовательности, соответствующие различным видам активности (ходьба, бег, падение и т.п.) и отклонения от нормы [11]. Для текстовых и логированных данных (журналы событий, отчёты о действиях) задействуются большие языковые модели: они рассматривают последовательности записей как естественный язык и способны выявлять аномалии или критические события по контексту [12, 13].

Рассмотренные нейросетевые подходы уже находят применение в ряде областей. Промышленное производство: большие модели компьютерного зрения используются для контроля действий работников на конвейерных линиях и обеспечения соблюдения техники безопасности. Например, системы на основе глубоких сетей в реальном времени выявляют отсутствие каски или другого средства защиты у сотрудника [14], что позволяет мгновенно реагировать и предотвращать инциденты. Кроме того, анализ вибраций и других сенсорных данных станков с помощью рекуррентных нейросетей помогает реализовать предиктивное обслуживание оборудования – обнаруживать отклонения в работе

механизмов и предупреждать аварии [15]. Умные дома: в бытовой среде крупные модели помогают мониторить повседневную активность жителей для повышения удобства и безопасности. Так, анализ данных камер и датчиков движения позволяет определить, что пожилой человек упал, и автоматически вызвать помощь. Носимые устройства (фитнес-браслеты, смарт-часы), оснащённые моделями распознавании человеческой деятельности (Human Activity Recognition, HAR), отслеживают показатели активности и здоровья пользователя, отправляя уведомления при выявлении аномального поведения (длительная неподвижность, аритмия и пр.) [16]. Безопасность и предотвращение рисков: нейросетевые модели активно внедряются в системы видеонаблюдения для распознавания подозрительных действий и ситуаций. С их помощью можно обнаружить на улице оставленный без присмотра предмет или агрессивное поведение в толпе и тем самым предупредить правонарушение. В кибербезопасности модели NLP анализируют сетевые логи и сообщения на наличие характерных паттернов, предшествующих атакам, позволяя оперативно реагировать на киберинциденты [17]. Ещё одним направлением применения крупных моделей является медицина и здоровье: обработка потоков данных от носимых сенсоров и даже анализ речи/текста пациентов (записи сессий, соцсети) с помощью LLM дают возможность выявлять признаки стресса, депрессии или ухудшения физического состояния на ранних стадиях [18, 2]. Таким образом, индустриальные кейсы демонстрируют универсальность больших нейросетевых моделей: они успешно работают от заводских цехов до домашних условий, повышая эффективность мониторинга и снижая фактор человеческой ошибки.

Постановка задачи

Цель: разработать прототип интеллектуальной системы, способной на локальной машине анализировать мультимодальные данные активности (видео, сенсоры, текстовые логи). Выяснить пригодность, точность и применимость использования больших предобученных нейросетевых моделей для отслеживания разных видов активности с целью дальнейшего взаимодействия на события.

Применение Ollama

В рамках реализации прототипа системы генерации текстов с использованием LLM применяется инструмент Ollama — программная платформа, позволяющая локально запускать и взаимодействовать с нейросетевыми моделями. Поддерживаются модели, основанные на современных архитектурах, таких как LLaMA, Mistral и других. Одним из ключевых преимуществ Ollama является возможность прямого

взаимодействия с моделью через REST API, что обеспечивает гибкость интеграции и контроль над параметрами генерации [19].

Взаимодействие с моделью осуществляется путём отправки HTTP-запросов на локальный сервер Ollama, который по умолчанию работает на порту 11434. Запросы формируются в формате JSON, где указываются как обязательные параметры (model, prompt), так и дополнительные настройки, влияющие на поведение генерации. На рис. 1 приведен фрагмент запроса к Ollama [19].

```
{
 "model": "llama3",
 "prompt": "Опишите роль нейросетей в современных производственных системах.",
 "temperature": 0.7,
 "format": "json",
 "stream": false
}
```

Рис. 1. Фрагмент запроса к Ollama

В запросе, приведенном на рис. 1, используются следующие поля:

- model – идентификатор используемой модели;
- prompt – текст запроса (инструкции), передаваемый в модель;
- temperature – параметр стохастичности генерации (в диапазоне от 0 до 1), влияющий на креативность ответа;
- format – формат выходных данных (text или json);
- stream – логический параметр, определяющий режим получения ответа (потоковый или полный) [19].

Prompt (далее - промпт) является важнейшей частью для построения верного запроса к модели. Без четкого и внятного промпта, модель может выдать неверные и непредсказуемые ответы, существенно снижая качество работы системы [20].

Для эффективной интеграции приложения Ollama в разрабатываемую систему была применена официальная Python-библиотека [21], предоставляющая интерфейс высокого уровня для отправки запросов и получения ответов от модели. На рис. 2 представлен блок кода взаимодействия с моделью на языке Python:

```python
import ollama
response = ollama.generate(
    model='llama3',
    prompt='Назовите ключевые принципы устойчивости нейронных сетей.',
    options={
        'temperature': 0.5,
        'format': 'json',
        'stream': False
    }
)
print(response['response'])
```

Рис. 2. Запрос к Ollama в Python

На рис. 3 приведен полученный результат, представляющий собой словарь, содержащий поля с метаданными, а также поле response, содержащее сгенерированный моделью текст.


```
{
 "model": "llama3",
 "created_at": "2025-03-24T12:34:56Z",
 "response": "Ключевыми принципами устойчивости нейронных
сетей являются способность к обобщению, толерантность к
шуму, адаптивность и интерпретируемость архитектуры.",
 "done": true
}
```


Рис.3. Ответ модели

Также предусмотрена возможность включения дополнительных параметров, позволяющие более тонко настраивать поведение модели:

- top_p – параметр выборки по вероятностному порогу (nucleus sampling);
- num_ctx – максимальное количество токенов контекста;
- repeat_penalty – штраф за повторение одинаковых токенов;
- stop – список токенов-стопов, при достижении которых генерация прекращается [19].

Техническая реализация

Система реализована в виде прототипа с использованием инструмента Ollama для запуска LLM локально на персональном компьютере.
На вход моделям подаются следующие данные:

- Видеопотоки (набор изображений с систем видеонаблюдения)
- Сенсоры (датчики температуры и влажности),
- Логи действий (формат JSON/текст).

Весь анализ будет проходить в несколько этапов.

Этап 1. Подготовка тестовых данных.

Перед тем как проводить анализ, необходимо подготовить набор входных данных для моделей. В качестве видеопотока с камер видеонаблюдения в целях защиты конфиденциальных данных используются изображения, сгенерированные нейросетью OpenAI ChatGPT-4o-mini [22], представленные ниже:

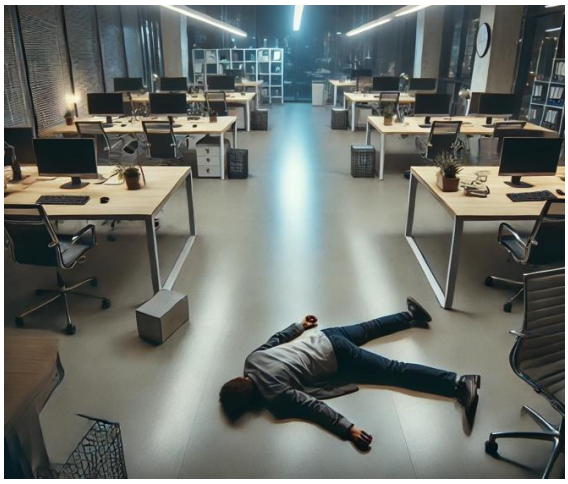

Рис. 4. Сгенерированное фото с камеры видеонаблюдения. Человек упал. Зафиксирована аварийная ситуация.

Рис. 5. Сгенерированное фото с камеры видеонаблюдения. Пустой коридор в офисе. Система видеонаблюдения не обнаружила нарушений.

Для имитации показаний датчика температуры и влажности используется пара случайных значений temperature и humidity, представленных в JSON-формате. В поле temperature выражается температура помещения в градусах по Цельсию, в поле humidity относительная влажность в процентах. Ниже представлены значения датчика в разное время, использовавшиеся для анализа.

```
{
 "temperature": 26.8,
 "humidity": 33.0
}
```


Рис. 6. Показания с датчика температуры и влажности. Нормальные показатели температуры и влажности.


```
{
 "temperature": 50.4,
 "humidity": 2.0
}
```


Рис. 7. Показания с датчика температуры и влажности. Сильное превышение температуры и понижение влажности. Вероятная причина - пожар.

Для имитации значений с логов прибегнем к журналированию любой другой уже существующей системы, которая может быть использована в контексте нашей задачи. К примеру, это могут быть СКУД (Система Контроля и Управления Доступом), охранные системы и т. д. Ниже приведен фрагмент лога со СКУД:


```
{
 "timestamp": "2025-03-29T13:19:00Z",
 "event": "Открыта входная дверь"
}
```


Рис. 8. Лог со СКУД. В данном примере поле timestamp означает время события в формате ISO 8601 [23], а event - описание самого события.

Этап 2. Выбор моделей и разработка скрипта анализа.

Для решения задачи были выбраны модели: gemma3:12b [24], llava:13b [25], llama3.2-vision:11b [26] и minicpm-v:8b [27]. Выбор этих моделей основан на том, что они являются наиболее популярными в контексте Ollama, а также имеют возможность анализа изображений [8].

Для анализа разработаем несколько тестовых сценариев, по которым будет определяться точность работы моделей. Сценарии представляют собой комбинацию из данных видео, датчиков и логов. Для сценариев используется заранее подготовленные данные на этапе 1.

Сценарий 1. Человек упал. Система анализирует фото с камеры на рис. 4 и набор параметров датчиков на рис. 6. В этом сценарии модель должна проанализировать на фото, что человек лежит без сознания на полу и сообщить, что произошла критическая ситуация, и нужна реакция.

Сценарий 2. Пожар с людьми в помещении. Система анализирует фото с камеры на рис. 4 и набор параметров датчиков на рис. 7. Также есть логи о пожаре от системы безопасности. Здесь модель должна проанализировать логи о пожаре, принять во внимание информацию от датчика, что температура повышена, и что на кадрах видеокамеры есть люди. В результате модель должна сообщить, что произошла критическая ситуация, люди находятся в опасности, и нужна реакция.

Сценарий 3. Пожар без людей в помещении. Система анализирует фото с камеры на рис. 5 и набор параметров датчиков на рис. 7. Сценарий схож со сценарием 2, за исключением того, что на кадрах нет людей. Здесь модель должна также сказать, что произошла критическая ситуация, но на кадрах нет людей, поэтому пожар не представляет опасности, соответственно модель не должна реагировать.

Сценарий 4. Штатный режим. Система анализирует фото с камеры на рис. 5 и набор параметров датчиков на рис. 6. На фото изображен пустой коридор, все показатели датчиков в норме. Логов от других систем нет. Модель не должна реагировать.

Для проработки данных сценариев был разработан скрипт на языке программирования Python. Скрипт, исходя из сценария, подставляет в запрос необходимые показания датчиков, логи с других систем и изображения с камер, находящиеся в папке img. Он прогоняет по каждую заранее выбранную модель по каждому из сценариев, замеряет время исполнения и записывает ответы в папку out.

Для моделей также объявлен параметр temperature = 0 для более детерминированных ответов, а также в format указана ожидаемая на выходе JSON-структура, где содержится два параметра:

- need_help - boolean, означает - нужна ли реакция;
- message - string, текстовое представление ситуации.

Этап 3. Подсчет результатов.

Для подсчетов результатов правильности используется функция F1-мера (F1-Score), Точность (Precision) и Полноту (Recall) [28], находящееся в пакете scikit-learn [29]. Ниже представлена таблица соотношения модели и их показателей по всем сценариям.

Модель	F1-мера (F1-Score)	Точность (Precision)	Полнота (Recall)
llama3.2-vision:11b	0.50	0.50	0.50
gemma3:12b	0.8	1.00	0.67
llava:13b	0.0	0.00	0.00

minicpm-v:8b	0.67	1.00	0.50

Таблица 1. Модели и их показатели

Другим важным параметром является время ответа модели. Ниже представлена таблица соотношения модели и затраченного времени на каждый сценарий.

Модель	Сценарий 1	Сценарий 2	Сценарий 3	Сценарий 4
llama3.2-vision:11b	26,19 с	3,74 с	23,75 с	1,84 с
gemma3:12b	14,5 с	11,1 с	10,7 с	10,8 с
llava:13b	15,54 с	6,13 с	5,42 с	4,64 с
minicpm-v:8b	9,95 с	1,69 с	7,50 с	1,67 с

Таблица 2. Время ответа моделей на каждый сценарий

Исходя из полученных данных можно заметить, что модель gemma3:12b оказалась наиболее точной по сравнению с другими, однако время исполнения всех запросов данной модели оказалась наибольшей. Модель смогла ответить правильно на 3 из 4 сценариев. Наихудшей оказалась модель llava:13b, не ответив верно ни на один сценарий.

Весь код скриптов, изображения, а также результаты моделей опубликованы на GitHub: https://github.com/minneakhmetov/llm-activity-abrau.

Заключение

Разработанный прототип демонстрирует перспективность применения больших нейросетевых моделей в системах отслеживания активностей. Эксперимент показал, что заранее предобученные на больших данных модели пригодны для решения задачи работы. Использование Ollama позволяет запускать LLM локально, что особенно важно при работе с чувствительными данными, однако все еще стоит вопрос о повышении быстродействия системы. В будущем возможно расширение за счёт онтологической поддержки сценариев и оптимизации на слабых устройствах.

Литература

1. Onsu M.A., Lohan P., Kantarci B., Syed A., Andrews M., Kennedy S. Leveraging Multimodal-LLMs Assisted by Instance Segmentation for Intelligent Traffic Monitoring [Электронный ресурс] // arXiv. — 2025. — URL: https://arxiv.org/abs/2502.11304 (дата обращения: 15.05.2025).

2. Ferrara E. Large Language Models for Wearable Sensor-Based Human Activity Recognition, Health Monitoring, and Behavioral Modeling // Sensors. — 2024. — Т. 24, № 15. — С. 5045.
3. Suh S., Rey V.F., Lukowicz P. Tasked: Transformer-based adversarial learning for human activity recognition using wearable sensors // Knowledge-Based Systems. — 2023. — Т. 260. — С. 110143.
4. Котенко И.В., Полубелова О.В., Саенко И.Б., Чечулин А.А. Применение онтологий и логического вывода для управления информацией и событиями безопасности // Системы высокой доступности. — 2012. — Т. 8, № 2. — С. 100–108.
5. Nour B., Pourzandi M., Debbabi M. A Survey on Threat Hunting in Enterprise Networks // IEEE Communications Surveys & Tutorials. — 2023. — Т. 25. — С. 2299–2324. — DOI: 10.1109/COMST.2023.3299519.
6. Николенко С.И. Машинное обучение: основы. — СПб.: Питер, 2025. — 608 с.
7. Bhardwaj A., Tripathi R., Bera P., Mavroeidis V., Gkioulos V. BTH: Behavior-based structured threat hunting framework to analyze and detect advanced adversaries // Electronics. — 2022. — Т. 11, № 19. — С. 2992. — DOI: 10.3390/electronics11192992.
8. Ollama: [Электронный ресурс]. — URL: https://ollama.com/ (дата обращения: 30.03.2025).
9. Nath N.D., Behzadan A.H., Paal S.G. Deep learning for site safety: Real-time detection of personal protective equipment // Automation in Construction. — 2020. — Т. 112. — С. 103085.
10. Gupta S. Deep learning-based human activity recognition using wearable sensor data // International Journal of Information Management Data Insights. — 2021. — Т. 1. — С. 100046.
11. Uçar A., Karakoşe M., Kırımça N. Artificial Intelligence for Predictive Maintenance Applications: Key Components, Trustworthiness, and Future Trends // Applied Sciences. — 2024. — Т. 14, № 2. — С. 898.
12. Пятаева А.В., Мерко М.А., Жуковская В.А., Казакевич А.А. Распознавание активности человека по видеоданным // International Journal of Advanced Studies. — 2022. — Т. 12, № 4. — С. 96–110.
13. Han S., Yuan S., Trabelsi M. LogGPT: Log Anomaly Detection via GPT [Электронный ресурс] // arXiv. — 2023. — URL: https://arxiv.org/pdf/2309.14482 (дата обращения: 15.05.2025).
14. Özüağ S., Ertuğrul Ö. Enhanced Occupational Safety in Agricultural Machinery Factories: Artificial Intelligence-Driven Helmet Detection Using Transfer Learning and Majority Voting // Applied Sciences. — 2024. — Т. 14. — С. 11278. — DOI: 10.3390/app142311278.

15. Li X., Chen Y., Hu L. Real-time workplace activity recognition using deep learning models // IEEE Transactions on Industrial Informatics. — 2023. — Т. 19, № 2. — С. 1520–1532.
16. Wu Z., Zhao J., Shen H. Smart home automation based on human activity recognition: A survey // Future Generation Computer Systems. — 2023. — Т. 137. — С. 41–57.
17. Sharma R., Patel N. Deep learning-based anomaly detection in surveillance videos // Journal of Visual Communication and Image Representation. — 2022. — Т. 86. — С. 103624.
18. Yadav S., Jha C.K., Kumar N. AI-powered fall detection systems for elderly care: Challenges and future directions // Computer Methods and Programs in Biomedicine. — 2024. — Т. 230. — С. 107416.
19. Ollama API Documentation: [Электронный ресурс]. — URL: https://github.com/ollama/ollama/blob/main/docs/api.md (дата обращения: 30.03.2025).
20. Sahoo P., Singh A.K., Saha S., Jain V., Mondal S., Chadha A. A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications [Электронный ресурс] // arXiv. — 2024. — URL: https://arxiv.org/pdf/2402.07927 (дата обращения: 15.05.2025).
21. Ollama Python Library: [Электронный ресурс]. — URL: https://github.com/ollama/ollama-python (дата обращения: 30.03.2025).
22. OpenAI ChatGPT-4o-mini: [Электронный ресурс]. — URL: https://chatgpt.com/ (дата обращения: 30.03.2025).
23. ISO 8601-1:2019 Standard: [Электронный ресурс]. — URL: https://www.iso.org/obp/ui/#iso:std:iso:8601:-1:ed-1:v1:en (дата обращения: 30.03.2025).
24. Ollama gemma3:12b Model: [Электронный ресурс]. — URL: https://ollama.com/library/gemma3:12b (дата обращения: 30.03.2025).
25. Ollama llava:13b Model: [Электронный ресурс]. — URL: https://ollama.com/library/llava:13b (дата обращения: 30.03.2025).
26. Ollama llama3.2-vision:11b Model: [Электронный ресурс]. — URL: https://ollama.com/library/llama3.2-vision (дата обращения: 30.03.2025).
27. Ollama minicpm-v:8b Model: [Электронный ресурс]. — URL: https://ollama.com/library/minicpm-v (дата обращения: 30.03.2025).
28. Hand D.J., Christen P. F*: an interpretable transformation of the F-measure // Journal of Classification. — 2021. — Т. 38, № 1. — С. 3–17.
29. Scikit Learn F1-Score: [Электронный ресурс]. — URL: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html (дата обращения: 30.03.2025).

References

1. Onsu M.A., Lohan P., Kantarci B., Syed A., Andrews M., Kennedy S. Leveraging Multimodal-LLMs Assisted by Instance Segmentation for

Intelligent Traffic Monitoring [Electronic resource] // arXiv. — 2025. — URL: https://arxiv.org/abs/2502.11304 (accessed: 15.05.2025).
2. Ferrara E. Large Language Models for Wearable Sensor-Based Human Activity Recognition, Health Monitoring, and Behavioral Modeling // Sensors. — 2024. — Vol. 24, No. 15. — P. 5045.
3. Suh S., Rey V.F., Lukowicz P. Tasked: Transformer-based adversarial learning for human activity recognition using wearable sensors // Knowledge-Based Systems. — 2023. — Vol. 260. — P. 110143.
4. Kotenko I.V., Polubelova O.V., Saenko I.B., Chechulin A.A. Primenenie ontologii i logicheskogo vyvoda dlya upravleniya informatsiei i sobytiyami bezopasnosti // Sistemy vysokoi dostupnosti. — 2012. — Vol. 8, No. 2. — P. 100–108.
5. Nour B., Pourzandi M., Debbabi M. A Survey on Threat Hunting in Enterprise Networks // IEEE Communications Surveys & Tutorials. — 2023. — Vol. 25. — P. 2299–2324. — DOI: 10.1109/COMST.2023.3299519.
6. Nikolenko S.I. Mashinnoe obuchenie: osnovy. — SPb.: Piter, 2025. — 608 p.
7. Bhardwaj A., Tripathi R., Bera P., Mavroeidis V., Gkioulos V. BTH: Behavior-based structured threat hunting framework to analyze and detect advanced adversaries // Electronics. — 2022. — Vol. 11, No. 19. — P. 2992. — DOI: 10.3390/electronics11192992.
8. Ollama: [Electronic resource]. — URL: https://ollama.com/ (accessed: 30.03.2025).
9. Nath N.D., Behzadan A.H., Paal S.G. Deep learning for site safety: Real-time detection of personal protective equipment // Automation in Construction. — 2020. — Vol. 112. — P. 103085.
10. Gupta S. Deep learning-based human activity recognition using wearable sensor data // International Journal of Information Management Data Insights. — 2021. — Vol. 1. — P. 100046.
11. Uçar A., Karakoşe M., Kırımça N. Artificial Intelligence for Predictive Maintenance Applications: Key Components, Trustworthiness, and Future Trends // Applied Sciences. — 2024. — Vol. 14, No. 2. — P. 898.
12. Pyatayeva A.V., Merko M.A., Zhukovskaya V.A., Kazakevich A.A. Raspoznavanie aktivnosti cheloveka po videodannym // International Journal of Advanced Studies. — 2022. — Vol. 12, No. 4. — P. 96–110.
13. Han S., Yuan S., Trabelsi M. LogGPT: Log Anomaly Detection via GPT [Electronic resource] // arXiv. — 2023. — URL: https://arxiv.org/pdf/2309.14482 (accessed: 15.05.2025).
14. Özüağ S., Ertuğrul Ö. Enhanced Occupational Safety in Agricultural Machinery Factories: Artificial Intelligence-Driven Helmet Detection Using Transfer Learning and Majority Voting // Applied Sciences. — 2024. — Vol. 14. — P. 11278. — DOI: 10.3390/app142311278.

15. Li X., Chen Y., Hu L. Real-time workplace activity recognition using deep learning models // IEEE Transactions on Industrial Informatics. — 2023. — Vol. 19, No. 2. — P. 1520–1532.
16. Wu Z., Zhao J., Shen H. Smart home automation based on human activity recognition: A survey // Future Generation Computer Systems. — 2023. — Vol. 137. — P. 41–57.
17. Sharma R., Patel N. Deep learning-based anomaly detection in surveillance videos // Journal of Visual Communication and Image Representation. — 2022. — Vol. 86. — P. 103624.
18. Yadav S., Jha C.K., Kumar N. AI-powered fall detection systems for elderly care: Challenges and future directions // Computer Methods and Programs in Biomedicine. — 2024. — Vol. 230. — P. 107416.
19. Ollama API Documentation: [Electronic resource]. — URL: https://github.com/ollama/ollama/blob/main/docs/api.md (accessed: 30.03.2025).
20. Sahoo P., Singh A.K., Saha S., Jain V., Mondal S., Chadha A. A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications [Electronic resource] // arXiv. — 2024. — URL: https://arxiv.org/pdf/2402.07927 (accessed: 15.05.2025).
21. Ollama Python Library: [Electronic resource]. — URL: https://github.com/ollama/ollama-python (accessed: 30.03.2025).
22. OpenAI ChatGPT-4o-mini: [Electronic resource]. — URL: https://chatgpt.com/ (accessed: 30.03.2025).
23. ISO 8601-1:2019 Standard: [Electronic resource]. — URL: https://www.iso.org/obp/ui/#iso:std:iso:8601:-1:ed-1:v1:en (accessed: 30.03.2025).
24. Ollama gemma3:12b Model: [Electronic resource]. — URL: https://ollama.com/library/gemma3:12b (accessed: 30.03.2025).
25. Ollama llava:13b Model: [Electronic resource]. — URL: https://ollama.com/library/llava:13b (accessed: 30.03.2025).
26. Ollama llama3.2-vision:11b Model: [Electronic resource]. — URL: https://ollama.com/library/llama3.2-vision (accessed: 30.03.2025).
27. Ollama minicpm-v:8b Model: [Electronic resource]. — URL: https://ollama.com/library/minicpm-v (accessed: 30.03.2025).
28. Hand D.J., Christen P. F*: an interpretable transformation of the F-measure // Journal of Classification. — 2021. — Vol. 38, No. 1. — P. 3–17.
29. Scikit Learn F1-Score: [Electronic resource]. — URL: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html (accessed: 30.03.2025).