

Логическая модель данных для систем интеллектуального видеонаблюдения

Н. А. Жукова

Санкт-Петербургский государственный
электротехнический университет
«ЛЭТИ» им. В.И. Ульянова (Ленина)
nazhukova@mail.ru

А. Н. Субботин

Санкт-Петербургский государственный
электротехнический университет
«ЛЭТИ» им. В.И. Ульянова (Ленина)
alesu1543@gmail.com

Аннотация. В статье представлена логическая модель данных для систем интеллектуального видеонаблюдения. Рассмотрены новые возможности по созданию моделей данных, ориентированные на широкое применение методов машинного обучения. Предложены новые способы построения логических моделей и их применения в системах интеллектуального видеонаблюдения. Разработано приложение для смартфона. Доказана эффективность использования предлагаемых логических моделей при обработке данных в туманных и облачных средах. Приводятся конкретные примеры.

Ключевые слова: модели данных; машинное обучение; туманные и облачные среды; обработка видеoinформации; Интернет Вещей

I. ВВЕДЕНИЕ

Большое значение для системы интеллектуального видеонаблюдения имеет система хранения данных. От системы хранения зависит быстрота обработки видеoinформации и точность определения событий. Возможность оперативного доступа к данным при их обработке с применением машинного обучения зависит от физических носителей и логической модели данных [1], [2].

Данные, как правило, хранятся в предобработанном виде в базе данных (БД). Применяемые способы хранения определяются используемыми техническими средствами и логической архитектурой, которая учитывает физические особенности технических средств. Применение MySQL предполагает использование следующих стандартов хранения на физическом уровне: InnoDB, MyISAM, Aria и ряда других. Необходимо учитывать физическую кластеризацию накопителя информации. Производитель серверного оборудования Cisco учитывает особенности физического хранения и предоставляет все необходимые драйвера.

Логическая архитектура системы хранения информации напоминает слоеный пирог, в котором на первом уровне – тип базы данных и система управления БД (СУБД), на втором уровне – стандарты физического уровня, далее все необходимые драйвера и средства оптимизации хранения в чипах устройства.

Хранение видеoinформации для систем интеллектуального видеонаблюдения имеет свои особенности. Видеoinформация представляется в виде

серии кадров и имеет большой объем. Однако, для алгоритмов машинного обучения не требуется обладать всей информацией: цветностью, прозрачностью, также не требуется хранить мелкие детали. Видео разделяется на серию кадров (например, 10 кадров в секунду, но не более 24), которые сохраняются в озере данных. Схема хранения данных для систем интеллектуального видеонаблюдения представлена рис. 1, где управляющая программа распределяет данные между базой данных (БД) и озером данных. База данных содержит три уровня: физический и драйвера, стандарты хранения с кластеризацией под типы данных и непосредственно сама БД с СУБД.

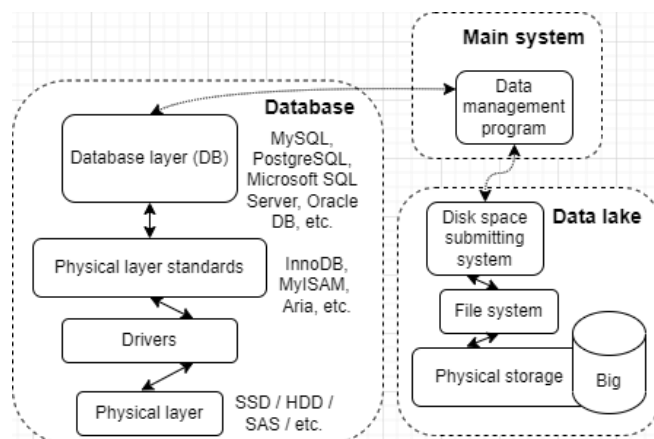


Рис. 1. Схема хранения данных

СУБД, в свою очередь, делятся на браузерные (которые можно открыть в браузере) и прикладные (в виде приложений к операционной системе). К первым относятся: phpMyAdmin, Oracle DB Admin и пр.; ко вторым: Microsoft SQL Management Studio, pgAdmin и пр. СУБД имеют большое значение при отладке и разработке программ, которые взаимодействуют с БД.

II. МОТИВАЦИЯ И ОПИСАНИЕ ПРОБЛЕМЫ

Системы интеллектуального видеонаблюдения сохраняют данные в файловой системе, при этом формируется структура, включающая различные каталоги, которые содержат текстовые файлы, Excel, Word, фрагменты видео, серии кадров и пр. Возникает существенная проблема в управлении хранением информацией, по причине её неоднородности (рис. 2) [3], [4].



Рис. 2. Проблема в управлении хранением информации

Управление осуществляется с использованием драйвера FS операционной системы. Отсутствие систематизации, организованного хранения приводит к низкой скорости доступа к данным и низкой скорости определения событий в системе интеллектуального видеонаблюдения [5].

III. СПОСОБ РЕШЕНИЯ

Предлагается раздельное хранение данных для повышения скорости чтения и записи данных, при этом способ хранения зависит от типа информации [6].

В одних случаях, предпочтительным является хранение в БД, а в других – в файловой системе. Хранение текстовой информации не в файлах, а в строках БД, позволяет существенно повысить скорость записи и чтения, поскольку чтение текстовых файлов занимает значительное время. Хранение таблиц не в Excel, а в БД обеспечивает повышение скорости работы при обращении к табличным данным [7].

Хранение информации в текстовых файлах и Excel, Word в настоящее время не вызывает затруднений, поскольку предоставляется программный интерфейс, обеспечивающий возможность доступа к данным из программных систем, разрабатываемых в различных средах (Visual Studio, RAD Studio, Qt и пр.) с применением многих библиотек (Developer Express, Fast Reports и пр.), которые позволяют не только записывать и читать данные из Excel, но и заполнять таблицы, которые могут быть использованы как из среды Visual Studio, так и при создании программ на PHP/JS для сайтов.

Для интеллектуализации системы хранения данных необходима единая программа управления потоками информации [8], [9]. В качестве технологии была выбрана XAMPP 8.1.2 (<https://www.apachefriends.org>), которая объединяет в себе web-технологии: Apache 2.4.39, MariaDB 10.3.16, PHP 7.3.6 и панель управления XAMPP v.3.2.4, применяемые для создания web-приложений, не отличающихся по функциональности от настольных бинарных приложений (desktop application) для ОС Windows, Linux, macOS, FreeBSD, QNX, OS/2 и пр.

В качестве приложения для контроля и управления хранением данных может использоваться не только web-сайт, но и приложение для смартфона на основе PWA (прогрессивное web-приложение), которое напоминает мобильное приложение для смартфон (*.ipa, *.apk и пр.), но использует смешанный web-контент [10], [11]. Такие приложения напоминают классические бинарные приложения, созданные с компонентом «browser», но применяют совершенно другую технологию, которая позволяет отправлять информацию в строку состояний смартфона, упрощает взаимодействие с сайтом через отдельное приложение, которое может быть также открыто в браузере (Opera Mini, Mozilla Firefox и пр.).

ТАБЛИЦА I ПЛАНИРУЕМЫЕ ПОКАЗАТЕЛИ

№	Показатели эффективности	
	Название показателя	Планируется
1.	Сокращение времени доступа к данным ^а	в 2 раза
2.	Повышение точности определения образов и событий	10%
3.	Повышение скорости принятия решений	25%

^а. Главный показатель. (Основной)

Идентификация образов осуществляется на облачном сервере Microsoft Azure (<https://azure.microsoft.com>) в отдельной программе, которая получает в качестве входных данных серию кадров (10 в 1 секунду) из туманной среды и идентифицирует образы с применением заранее построенных моделей, для создания которых использовался язык Python и библиотека TensorFlow (<https://www.tensorflow.org>). Точность определения образов может быть повышена за счет увеличения частоты кадров, передаваемых в облако, что становится возможным благодаря повышению скорости записи и чтения данных из хранилища [12]. Остальные показатели являются взаимозависимыми.

IV. ПРИМЕР ИСПОЛЬЗОВАНИЯ

Для оценки эффективности предложенной логической модели данных разработано три приложения для управления данными:

- приложение для desktop с возможностью работы под Linux;
- web-сайт, который работает в любом современном браузере и операционной системе на основе двух стандартов: HTML5 и CSS 3.0;
- PWA-приложение под смартфон HTC, которое совмещает элементы web-сайта и классического приложения под Android или iOS.

Приложение (рис. 3) со стандартным интерфейсом VCL Application для рабочего стола KDE/Gnome разработано в Lazarus 2.0.10 с возможностью компиляции в Embarcadero RAD Studio 10.4 с применением компонента «DAC for MySQL» (Direct Access Components for MySQL) от компании Microolap (<https://www.microolap.com>), что позволяет повысить скорость работы при компиляции под Windows 10/11.

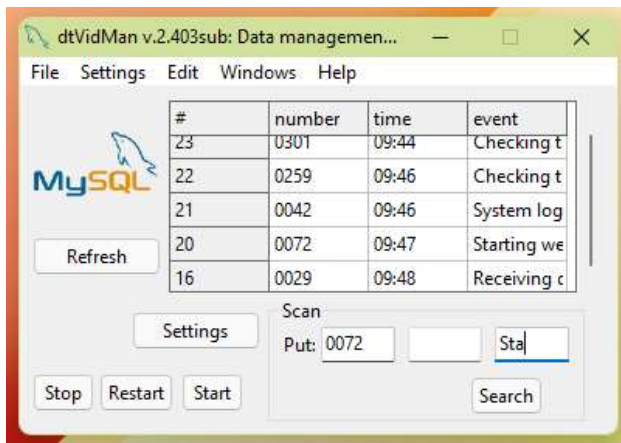


Рис. 3. Внешний вид десктоп приложения (скомпилированного под Windows 11)

Web-приложение для управления данными (рис. 4), разработанное с применением технологий PHP и JavaScript, позволяет показывать статистику в удобном табличном виде, где в первом столбце указывается номер диагностического сообщения, во втором – время события. Третий столбец в таблице является основным и содержит описание произошедшего события.

В приложении имеется основное меню, которое позволяет добавлять события в режиме тестирования с правами «root», выполнять подготовку таблицы событий без стилей для печати и осуществлять переход в корневой каталог сайта.



Рис. 4. Внешний вид web-сайта

Приложение под смартфон HTC (рис. 5) предусматривает отображение статистики управления данными в центральной части экрана, имеет возможность остановки по кнопке STOP, перезапуска по кнопке RESTART и запуска приложения, если оно не было ранее запущено. Для работы с приложением, необходимо установить «dtVideoManager-0.402a.apk» из файловой системы смартфона, которое может работать под любой Android-совместимой операционной системой Linux.



Рис. 5. Внешний вид приложения для смартфона HTC

Для проведения замеров времени доступа к данным в рамках оценки первого показателя эффективности был сформирован массив данных объемом 30 Мбайт. Среднее время доступа сократилось в 2,368 раза, а время доступа к текстовым файлам почти в 5 раз (4,86).

ТАБЛИЦА II СОКРАЩЕНИЕ ВРЕМЕНИ ДОСТУПА К ДАННЫМ

№	Тип данных	Снижение времени доступа (мсек., в разы)		
		Без применения логической модели	С применением логической модели	Соотношение
1.	Текстовые файлы ^а	628	129	4,86
2.	Видео *.MPEG4, *.AVI	5289	3721	1,42
3.	Таблицы Excel	2647	1682	1,57
4.	Файлы Word	1372	783	1,75
5.	Кадрированные изображения	782	348	2,24
	Среднее:			2,368

^а Главный показатель. (текст)

Вторым показателем эффективности является точность определения событий с применением методов машинного обучения. Для его оценки были проведены эксперименты в период с 18 февраля 2022 года по 3 марта 2022 года в облаке для определения событий на объектах метрополитена. Средняя точность определения событий составила 10,8 %.

ТАБЛИЦА III Точность определения событий

№	Событие	Определение событий (%)		
		Без применения логической модели	С применением логической модели	Соотношение
1.	Пожар ^а	0.85	0.96	11%
2.	Ограбление	0.83	0.92	9%
3.	Затопление	0.79	0.94	15%
4.	Террористическая угроза	0.87	0.98	11%
5.	Световое излучение	0.89	0.97	8%
	Среднее:			10,8%

^а Главный показатель. (пожар)

По третьему показателю эффективности (скорость принятия решений) были получены наиболее высокие оценки. При его оценке использовалось разработанное приложение для смартфона HTC в облаке для оценки скорости принятия решений на объектах метрополитена. Полученные результаты показали, что скорость принятия решений увеличилась на 27,4 %.

ТАБЛИЦА IV СКОРОСТЬ ПРИНЯТИЯ РЕШЕНИЙ

№	Принятие решения (сек, %.)			
	Служба	Без применения логической модели	С применением логической модели	Соотношение
1.	Пожарная ^d	67	52	22,38%
2.	Полиция	239	183	23,43%
3.	Водопроводчики	692	517	25,3%
4.	ФСБ	948	702	25,94%
5.	Электрики	1482	891	39,87%
	Среднее:			27,4%

^d. Главный показатель. (пожар)

Для повторения исследования необходимо установить программы для визуализации статистики и управления хранением данных, накопить статистику за определенное время. Для этого необходимо разработать программы для агрегации статистики или сохранить файлы логирования системы интеллектуального наблюдения в Libre Calc, применить фильтрацию, выполнить операции: агрегирование, суммирование, расчет среднего значения, и заполнить результирующие таблицы.

Эксперименты проводились на смартфоне HTC One E8 DS 16 Гб с техническими характеристиками: 4x2.3 ГГц, 2 Гб, 2 SIM, Super LCD3, 1920x1080, 4G, GPS, FM, microSD. Для сбора статистики применялся ноутбук Asus E510 с техническими характеристиками: Intel Celeron N4020 2.8 ГГц, Intel UHD Graphics 600, 8Гб DDR4, eMMC 256Гб.

V. ЗАКЛЮЧЕНИЕ

В статье рассматривается логическая модель данных для систем интеллектуального видеонаблюдения. Предлагается раздельное хранение информации в зависимости от типа (текст, Excel, видео, Word, серия кадров). Обосновывается целесообразность в качестве системы хранения данных использовать СУБД MySQL и файловую систему под Linux. Показана необходимость управления хранением данных. Представлены разработанные приложения для сбора статистики и управления хранением данных.

Анализ результатов проведенных экспериментов позволил подтвердить эффективность применения предложенной логической модели: обеспечено сокращение времени доступа к данным более, чем в 2 раза, повышение точности определения событий в среднем почти на 11 %, скорости принятия решений на более, чем 27 %. Представленная модель может применяться в различных общественных учреждениях: в метрополитене, больницах, ресторанах, стадионах и концертных площадках.

БЛАГОДАРНОСТЬ

Авторы выражают благодарность за помощь в подготовке и проведении исследования ГУП «Петербургский метрополитен» (<http://www.metro.spb.ru>) и ФИЦ РАН (<https://spcras.ru/spiiras>), за научную поддержку, помощь в подготовке статьи и доклада на конференции SCM-2022 в Санкт-Петербурге, Россия.

СПИСОК ЛИТЕРАТУРЫ

- [1] Mahmood Z., Guide to Ambient Intelligence in the IoT Environment Principles, Technologies and Application // Springer International Publishing AG, Cham, Switzerland, 2019. 289 p.
- [2] Muhammad Ali Babar Alan W. Brown Ivan Mistrik Agile Software Architecture Aligning Agile Processes and Software Architectures // Morgan Kaufmann, MA, USA, 2014. 292 p.
- [3] Gonçalo Marques, Rui Pitarna, Nuno M. Garcia and Nuno Pombo. Internet of Things Architectures, Technologies, Applications, Challenges, and Future Directions for Enhanced Living Environments and Healthcare Systems: A Review. Electronics 2019, 8, 1081; doi:10.3390/electronics8101081. www.mdpi.com/journal/electronics
- [4] Korzun D., Balandina E.; Kashevnik A.; Balandin S.; Viola F. Ambient Intelligence Services in IoT Environments: Emerging Research and Opportunities; IGI-Global, 2019, 199 p.; doi:10.4018/978-1-52258973-0.
- [5] Subbotin A.N. and Zhdanov V.S., "Application of Machine Learning Methods to Control the Process of Defectoscopy of Railway Tracks," 2021 IV International Conference on Control in Technical Systems (CTS), 2021, pp. 64-67, doi: 10.1109/CTS53513.2021.9562911.
- [6] Khaled Matrouk, Kholoud Alatoun Scheduling. Algorithms in Computing: A Survey. International Journal of Networked and Distributed Computing. Volume 9, Issue 1, January 2021, Pages 59-74. doi:10.2991/ijndc.k.210111.001.
- [7] Subbotin A.N., "Data Processing in Foggy Computing Environments for Machine Learning," 2021 II International Conference on Neural Networks and Neurotechnologies (NeuroNT), 2021, Pages 51-53. doi: 10.1109/NeuroNT53022.2021.9472203.
- [8] Amol Kulkarni, Janis Terpenny, Vittaldas Prabhu. Sensor Selection Framework for Designing Fault Diagnostics System. Sensors. Volume 21(19), 1 October 2021, 6470. doi:10.3390/s21196470.
- [9] Subbotin A.N., "Applying Machine Learning in Fog Computing Environments for Panoramic Teeth Imaging," 2021 XXIV International Conference on Soft Computing and Measurements (SCM), 2021, Pages 237-239. doi: 10.1109/SCM52931.2021.9507120.
- [10] Sanfelice R.G. Analysis and Design of Cyber-Physical Systems. A Hybrid Control Systems Approach // Cyber-Physical Systems: From Theory to Practice / D. Rawat, J. Rodrigues, I. Stojmenovic. CRC Press, 2016. ISBN 978-1-4822-6333-6.
- [11] Subbotin A., Zhukova N., Man T., "Architecture of the intelligent video surveillance systems for fog environments based on embedded computers," 2021 10th Mediterranean Conference on Embedded Computing (MECO), 2021, pp. 1-8. doi: 10.1109/MECO52532.2021.9460270.
- [12] Maria Gorlatova, Hazer Inaltekin, Mung Chiang. Characterizing task completion latencies in multi-point multi-quality computing systems. Computer Networks. Volume 181, 9 November 2020, 107526. doi:10.1016/j.comnet.2020.107526.