Федеральное государственное бюджетное учреждение науки

Вычислительный центр им. А.А. Дородницына

Российской академии наук

На правах рукописи

Абрамов Николай Александрович

ПРОГРАММНАЯ СИСТЕМА ВЫЯВЛЕНИЯ НЕЛЕГИТИМНОЙ АКТИВНОСТИ НА ПРОМЫШЛЕННЫХ ПЛОЩАДКАХ

Специальность 05.13.11 – Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей.

Диссертация на соискание ученой степени кандидата технических наук

Научный руководитель:

Доктор технических наук, профессор Дулин С.К.

Москва - 2013

Оглавление

Введение

Актуальность темы. В настоящее время появились новые информационные технологии, связанные с бихевиористическим анализом деятельности различных объектов. Эти технологии требуют сложного математического и программного обеспечения. Одной из областей бихевиористического анализа является разработка математических моделей и соответствующего программного обеспечения для компьютеризированных комплексов, предназначенных для автоматизированного анализа поведения различных объектов в заданных средах.

Исключительно большой практический интерес в бихевиористическом анализе имеет задача обнаружения нелегитимной активности на промышленных объектах. В настоящее время уровень математического и программного обеспечения такого анализа недостаточен для успешного решения практических задач в данной области. Фактически, системы, используемых для выявления нелегитимной активности, не имеют математической и программной составляющих, и сводятся к неавтоматизированному видеоконтролю территорий предприятий. В связи с этим разработка математического и программного обеспечения указанных бихевиористических технологий является чрезвычайно актуальной.

Цель данного диссертационного исследования – разработка математического и программного обеспечения информационной технологии для бихевиористического анализа потоков транспорта на предприятии в режиме реального времени.

Для достижения поставленной цели диссертационного исследования были решены следующие задачи:

- Разработана математическая модель классификации автотранспортных средств, позволяющая на основе анализа потока

видеоданных выявлять транспортные средства с признаками нелегитимной активности;

- Разработана архитектура программной системы бихевиористического анализа;
- Разработан программный модуль, реализующий интеграцию критериев в информационную систему;
- На базе разработанного математического обеспечения создан программно-аппаратный комплекс, реализующий систему анализа бихевиористического поведения.

Методы исследования. Теоретические и практические исследования базируются на методах распознавания образов (алгоритмы классификации, распознавание текстовых меток), системного программирования, методах построения вычислительных систем и математических методах моделирования нелегитимной активности.

Научная новизна. В диссертационной работе разработано новое математическое и программное обеспечение системы компьютеризированного выявления нелегитимной активности. Данная система основана на анализе признаков нелегитимной активности математическими методами.

Автором получены следующие результаты:

- Разработана математическая и программная модель обнаружения нелегитимной активности автотранспорта;
- На их основе разработано прикладное программное обеспечение, которое в комбинации с программным модулем распознавания автомобильных номеров без участия оператора с высокой вероятностью выявляет нелегитимную активность в режиме реального времени;
- Разработана архитектура программного комплекса для бихевиористического анализа данных видеонаблюдения;

- На основе анализа программных средств выявления нелегитимной активности разработаны требования к техническим параметрам проектируемой системы;
- Создано программно-прикладное средство управления обработкой данных: программный блок, реализующий признаки классификации в системе выявления нелегитимной активности;
- Разработана программная система, объединяющая распознавание и классификацию объектов бихевиористического анализа.

Практическая значимость работы заключается в создании промышленного образца системы безопасности и контроля транспортных потоков на производственных территориях. Продемонстрирована на практике эффективность методики выделения нелегитимной активности, исходя из шаблонов поведения. Функционирование системы «Цербер» привело к резкому снижению нелегитимной активности на Ижорской промышленной площадке. Разработанная программная система, обладая достаточной масштабируемостью, может быть использована и для контроля в крупных транспортных узлах.

На защиту выносятся следующие результаты:

- Разработанная математическая и программная модель выявления нелегитимной активности;
- Разработанная архитектура распределенной программной системы выявления нелегитимной активности на промышленных площадках;
- Разработанные программные блоки классификации нелегитимных событий;
- Программно-аппаратный комплекс «Цербер», реализующий разработанные математические модели и программные средства выявления нелегитимной активности в режиме реального времени.

Апробация. Основные положения диссертации докладывались на семинарах ИПИ РАН, а также ВЦ РАН в период с 2011 по 2013 годы. Результаты, полученные в ходе выполнения данной работы, вошли в ежегодные отчеты по проекту Российского фонда фундаментальных исследований № 11-07-00225 «Интеллектуализация методов описания геоинформационных объектов и технологических процессов». Также, результаты данной работы докладывались на IV Международной научной конференции «Фундаментальные проблемы системной безопасности и устойчивости», а также на Третьей Всероссийской научной конференции «Методы и средства обработки информации».

Публикации. По теме диссертационной работы опубликованы пять печатных работ [1-5], из них четыре в изданиях по перечню ВАК.

Структура диссертации. Диссертация состоит из введения, трех глав, заключения, списка литературы (наименований) и трех приложений. Работа изложена на 113 страницах, включающих 14 рисунков и 4 таблицы. Список использованной литературы включает в себя 79 наименований.

Глава 1.Обзор систем выявления нелегитимной активности

Эволюция систем видеонаблюдения

Использование систем видеонаблюдения на предприятиях началось с аналоговых систем CCTV (closed-circuit television – телевидение замкнутого контура). Эти системы состоят из нескольких камер, расположенных в разных местах наблюдаемой территории и связанных друг с другом набором мониторов, которые обычно помещаются в одну контрольную комнату в виде «матрицы мониторов». В настоящее время большинство систем CCTV используют аналоговые технологии для передачи и хранения изображений и цифровые ПЗС-матрицы для захвата изображений [37,40,68].

Работа систем первого поколения построена следующим образом – цифровое изображение с камер преобразуется в аналоговый композитный видеосигнал, который связан с мониторами и записывающим оборудованием, и передается, как правило, через коаксиальный кабель [74]. Существенным недостатком таких систем является ухудшение качества картинки из-за использования цифро-аналогового преобразования, что приводит к появлению «цифрового шума». Улучшение систем видеонаблюдения возможно за счет использования полностью цифрового формата для отснятых изображений и высокопроизводительных компьютеров для обработки.

Постепенное совершенствование технологии CCTV, привело к развитию полуавтоматических систем, называемых *вторым поколением систем видеонаблюдения*. Эти системы распознавали происходящие события на основе технологий машинного зрения. Они позволяли проводить примитивный анализ видеоряда и в полуавтоматическом режиме выявлять подозрительную активность. Обычно это производилось с помощью выделения подозрительных кадров и показа их оператору системы. Однако минусом таких системы были все же небольшое количество камер (в силу

несовершенства архитектуры) и слабые возможности интеллектуального анализа данных.

В настоящее время активно внедряются системы *третьего поколения*, основной отличительной чертой которых является иерархичность и распределение информационных потоков. Они могут состоять из огромного количества камер, а также использовать сложные алгоритмы машинного зрения, однако их возможности по автоматическому выявлению нелегитимных действий ограничены [45,46,70]. Большинство систем с высокой степенью автоматизации очень громоздки и сложны во внедрении. Сравнительные характеристики всех трех поколений систем приведены в таблице 1.1.

Таблица 1.1 Этапы технологического развития систем видеонаблюдения

Первое поколение	
Технология	**Аналоговые CCTV системы**
Преимущества	1. Хорошая производительность для простых задач 2. Проверенная временем технология
Проблемы	Аналоговые технологии для распространения и хранения данных
Текущее состояние систем	1. Цифровое или аналоговое 2. Цифровая запись 3. Сжатие видео

Второе поколение	
Технология	Автоматическое видеонаблюдение с использованием технологий машинного зрения
Преимущества	Повышает эффективность систем CCTV
Проблемы	Требуются надежные алгоритмы
Текущее состояние систем	1. Поиск надежных алгоритмов поведенческого анализа 2. Автоматическое обучение системы 3. Переход от статистического анализа сцены к пояснениям на естественном языке
Третье поколение	
Технология	Автоматическая система видеонаблюдения за большой территорией
Преимущества	1. Более точные за счет применения различных датчиков 2. Распределенность
Проблемы	1. Распределенность информации 2. Методология

	построения 3. Мультисенсорные платформы
Текущее состояние систем	1. Распределенный или централизованный «искусственный интеллект» 2. Объединение информации 3. Система принятия решений на основе статистики 4. Многокамерные способы наблюдения

Сферы применения систем видеонаблюдения

Особенно велика потребность в системах видеонаблюдения в следующих областях:

- Транспортные узлы, такие как: аэропорты, порты, вокзалы, метро и автомагистрали (для наблюдения за потоком машин).
- В местах большого скопления людей, таких как: учебные заведения, супермаркеты и автостоянки (для определения занятости мест).
- Спортивные сооружения, концертные залы (для обеспечения безопасности и учета числа свободных мест).
- Наблюдение за территорией на режимных объектах (для обеспечения безопасности).
- Контроль соблюдения технологической дисциплины в производственных процессах (для контроля персонала).

Кроме того, террористические акты, произошедшие в последние годы, привели к росту требований к обеспечению общественной безопасности. Одной из самых распространенных мер в этой области стало развертывание больших систем видеонаблюдения в местах, которые могут стать объектами атак террористов. Например, в лондонском метро и аэропорту Хитроу было установлено более 5000 камер [2,3,4]. Очевидно, что для эффективной обработки огромного потока информации, который поступает с такого числа камер, необходимо решить такие вопросы, как масштабируемость и удобство использования системы видеонаблюдения.

Подобные внедряемые системы видеонаблюдения, как правило, используют достаточно стандартное оборудование и в них все чаще применяются программные средства для обработки сигналов, поступающих из видеокамер [20,24,25]. Одной из основных задач для этих систем является распознавание несанкционированных вторжений и перемещений объектов, а также обнаружение подозрительных предметов.

Исследования в этой области [7,8,12,14,54,55,56,71] в основном сосредоточены на улучшении алгоритмов обработки изображений, что позволяет создавать новые методы для более точного и надежного обнаружения и распознавания объектов, слежения за ними, а также распознавания человеческой активности. Кроме того, проводятся разработки по созданию СУБД для систем видеонаблюдения и инструментов оценки эффективности систем видеонаблюдения. Также в настоящий момент исследуются новые способы для передачи видеоинформации в распределенных системах видеонаблюдения. Примерами таких задач являются методы сжатия видео, построение сетей и протоколов передачи данных, распределение обработки данных и стандарты форматов данных для передачи по сети.

Наиболее сложными и интеллектуальными являются системы распознавания выражений человеческих лиц, выявляющие потенциальных

преступников. Однако такие системы требуют большого количества камер, тщательной настройки и результаты их эксплуатации не всегда удовлетворительны [69,72,73].

Методики, применяемые в системах видеонаблюдения

Рассмотрим, как устроены современные системы видеонаблюдения. Типичная конфигурация модулей обработки видеоинформации в системе видеонаблюдения представлена на рисунке 1.1.

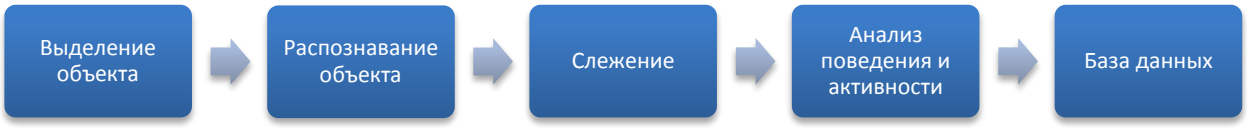


Рисунок 1.1 Типичная конфигурация блоков в современной системе видеонаблюдения.

Каждый из этих модулей представляет собой низкоуровневые блоки, которые есть в любой современной системе видеонаблюдения. Рассмотрим теперь более детально каждый из них.

Модуль выделения объектов

Существуют два классических подхода к выделению объектов: «временная разница» и «вычитание фона» [21,58]. После применения одного из этих подходов, применяются морфологические операции над изображением для снижения общего шума.

Первый подход заключается в вычитании двух последовательных кадров. Этот метод имеет хорошую производительность в средах с большим количеством движущихся объектов, в силу своей высокой адаптивности, но его недостаток состоит в том, что он плохо определяет все соответствующие объекту пиксели.

Второй метод основан на вычитании фона (или эталонной модели) из текущего изображения, он включает в себя создание фона модели и его динамическое обновление, для улучшения обнаружения объекта, когда

происходят изменения в окружающей среде. Метод вычитания фона имеет лучшую производительность для извлечения информации об объекте, но он весьма чувствителен к динамическим изменениям в окружающей среде.

Существуют различные подходы к моделированию фона, зависящие от сферы конкретного применения: например, в помещениях с хорошими условиями освещения и стационарными камерами, можно создать простой фон модели, сглаживая по времени последовательность полученных изображений за короткие промежутки времени.

При работе системы видеонаблюдения на открытом пространстве изменчивость фона, как правило, очень высока. При этом для работы системы необходимо иметь надежные адаптивные модели фона, что значительно повышает их вычислительную сложность. Типичным решением в этом случае является использование гауссовой модели (GM), в которой интенсивность каждого пикселя считается с помощью отдельного распределения Гаусса (GM-mixed). В рамках исследования [34] из-за особенностей окружающей среды (лес), используется комбинация двух моделей распределения Гаусса, чтобы справиться с бимодальностью фона (например, движением деревьев на ветру).

Авторы в статье [67] используют распределения Гаусса для моделирования каждого пикселя. При этом, медленному изменению освещения соответствует медленная адаптация параметров распределения Гаусса. Аналогичный метод используется в статье [51]. В работе [43] модель фона основана на оценке шума каждого пикселя в последовательности фоновых изображений, движущиеся объекты при этом определяются с помощью оценки шума, который порождает каждый пиксель.

Другие методы обнаружения объектов используют группы пикселей в качестве базовых объектов для отслеживания движений, группировка пикселей производится при помощи методов кластеризации, основанных на

информации о цвете пикселя (R, G, B) и его пространственном положении в двумерном пространстве (x, y), что делает кластеризацию более надежной.

Специальные алгоритмы, например, такие как минимизация ожидания (Expectation Minimization), используются для отслеживания движущихся объектов, сильно отличающихся от соответствующего эталонного образа, например в статье [34] авторы используют данный алгоритм, чтобы сгруппировать в кластеры траектории, относящиеся к одному типу движения, а затем выделить характерные особенности данного движения.

В статье [39] была изложена техника обнаружения объекта на основе вейвлет (wavelet) преобразования, коэффициенты которого использовались для определения вида пешеходов спереди и сзади. Используя HAAR-wavelet коэффициентов как низкоуровневых детекторов интенсивности изображения, можно извлечь информацию высокого уровня о выделяемом объекте (пешеходе) например, информацию о его форме. На этапе «обучения» системы, коэффициенты, которые наиболее точно описывают обнаруживаемый объект, выбираются с помощью больших обучающих наборов данных. Когда наилучшие коэффициенты были выбраны, используется система опорных векторов (SVM – support vector machine), для классификации обучающих наборов.

Работы системы состоит из двух этапов: на этапе обнаружения выбранных объектов сначала вычисляются коэффициенты из изображения, а затем SVM используется для проверки обнаружения объекта. Преимущество wavelet-методов состоит в том, что они не используют информацию о цвете или текстуре, поэтому они могут быть полезны в таких приложениях, где отсутствует информация о цвете (обычное явление при наблюдении внутри помещения). Более того, использование wavelet подразумевает значительное сокращение объема данных на этапе обучения. Тем не менее, авторы модели остановились только на проверке вида пешеходов спереди и сзади. В случае групп людей, которые останавливаются

или ходят перпендикулярно зрению камеры, алгоритм не срабатывает. Кроме того, возможно ложное срабатывание алгоритма на объектах с характеристиками движения аналогичными виду человека спереди или сзади.

Другое перспективное направление исследований [42] основано на обнаружении контуров лиц с использованием метода главных компонент (PCA – principal component analysis). При рассмотрении сегментации движения, полезными могут быть методы, основанные на анализе оптического потока, при этом система может использовать движущиеся камеры [35,36]. Следует отметить, что здесь могут возникать проблемы, когда размер отслеживаемых объектов мал, в связи с особенностями работы с небольшими областями изображения.

Распознавание объектов, отслеживание и оценка производительности

Существующие методы отслеживания можно разделить на три основных группы: 2D-модели с явной моделью формы, 2D-модели без явной модели формы и 3D-модели [41].

В работе [30] автор использует две 2D-модели для отслеживания автомобилей: прямоугольную модель для машины, проходящей близко от камеры и U-образную модель для машины, стоящей далеко от камеры (или наоборот – очень близко). Система состоит из блока получения изображения, модуля определения полосы движения и автомобиля, координатора процессов и нескольких регистраторов проезда автомобилей.

В некоторых многокамерных системах [23,76], акцент делается не самом объекте наблюдения, а на извлечении и последующем рассмотрении траекторий объектов, которые используются для построения геометрических и вероятностных моделей динамики.

Одним из наиболее важных вопросов для работы систем видеонаблюдения, является получение знаний о форме объектов. Априорное

знание о форме объекта может быть получено путем вычисления внешнего его вида в зависимости от его положения относительно камеры, сходным образом можно получить геометрию сцены. Однако для того чтобы создать форму модели, необходимы методы калибровки камер, обзор различных методов калибровки можно найти в работе [26].

После получения априорного знания, оно может быть использовано для создания надежного алгоритма отслеживания движения объектов, который будет способен эффективно работать в различных условиях, например, таких как изменение освещенности. Относительно несложно задать ряд ограничений на внешний вид объектов в модели, используя, например, тот факт, что люди ходят в вертикальном положении и в контакте с землей (данное ограничение может использоваться как в помещениях, так и вне помещений).

Система, представленная в работах [59] и [79], может распознавать и отслеживать автомобили с использованием 3D-модели транспортного средства, определяя его положение в плоскости земли и направление его движения. Также она способна распознавать и отслеживать пешеходов используя 2D-модели силуэта, основанной на контурах базисного сплайна.

Другой распространенный метод отслеживания движения состоит в использовании механизма фильтрации для прогнозирования каждого изменения положения объекта. Наиболее используемым для этих целей фильтром, является фильтр Калмана [1], т.н. оценивающий вектор динамической системы, являющийся одним из самых популярных алгоритмов фильтрации [11,60].

Еще одним методом, который часто используется в системах видеонаблюдения, является заполнение интересующей нас области изображения прямоугольниками или эллипсами (примитивами) В работе [48] автор отслеживает разные части человеческого тела с помощью примитивов, которые описываются в статистическом выражении в виде

пространственных и цветовых распределений Гаусса. Однако в некоторых ситуациях условия таковы, что невозможно применить линейный или фильтр Гаусса, и тогда применяют нелинейные байесовские фильтры, такие как, например расширенные фильтры Калмана (EKF) или фильтры частиц. В работе [67] было показано, что в сильно нелинейных средах фильтры частиц дают лучшие результаты, чем EKF.

Помимо использования подходов, основанных на получении априорного знания об объекте, существует другой подход [31], заключающийся в использовании компонент связности к сегменту, в котором происходят изменения. Этот подход имеет хорошую производительность, когда объект небольшой и его аппроксимируют с низким разрешением, а положение камеры тщательно выбрано. Примером данного подхода являются скрытые модели Маркова (HMM), которые используются, например, в работе [32], где авторы применяют расширение HMM для прогнозирования и отслеживания траектории объекта. Хотя HMM фильтры и пригодны для динамических сред (поскольку в такой модели нет ограничений на шум, требуемых для EKF), но модель сначала необходимо обучить в режиме оффлайн.

Отсюда можно сделать вывод о существовании большого количества различных методов фильтрации, каждый из которых может быть использован для определенных алгоритмов, поэтому для упрощения работы, были созданы специальные полуавтоматические системы, используемые для тестирования алгоритмов на заведомо верных данных [33].

Поведенческий анализ

Следующий этап обработки полученных с камер данных включает в себя распознавание системой видеонаблюдения поведения отслеживаемых объектов. Этот этап можно описать как решение проблемы классификации данных, имеющих временные метки. Данную задачу можно свести к сравнению текущей последовательности движений с предварительно

записанной библиотекой размеченных последовательностей, которые представляют собой прототипы различных действий. Есть несколько подходов для сравнения таких данных.

Например, метод Dynamic Time Warping (DTW) широко используется в распознавании речи и сравнении изображений [77], а также распознавании движений человека [61,76]. Хотя он очень надежен, но в настоящее время чаще используются HMM и байесовские сети [78], поскольку их проще реализовать, а результаты работы в целом сравнимы с DTW. Также не столь широко, как HMM, поскольку они не столь хорошо изучены для определения движения, но, тем не менее, применяются нейронные сети (NN). В работе [57] распознавание поведения осуществляется с использованием декларативной модели представления сценариев и формальной логики для определения предварительно записанных сценариев поведения. В работе [49] было предложено распознавать выражения лиц людей по кадрам и таким образом выявлять нелегитимную активность, однако получившаяся система достаточно громоздка и имеет большое количество ошибок второго рода [16,29,64,66].

База данных

Одним из заключительных этапов обработки в системе видеонаблюдения является хранение и поиск информации. Вопрос эффективного хранения и вывода информации разных форматов, и типов сравнительно мало исследовался. Однако и тут можно выделить несколько основных моделей. В статье [43] авторы определяют и создают модели данных для поддержки хранения информации такой системы на различных уровнях абстракции. В статье [44] авторы разработали модель данных и язык запросов для системы индексации и поиска данных для видео, которая позволяет вести поиск событий и объектов, а также вводить различные

ограничения. Извлечение данных выполняется на основе системы правил для чего система имеет декларативную и операционную семантику для того чтобы устанавливать связи между различными объектами системы. Видеоряд, полученный с камер, разбивается на множество фрагментов, каждый из которых может быть проанализирован на предмет извлечения информации (явные описания), которая может быть сохранена в базе.

В статье [50] также описывается методика извлечения информации на основе указанной классификации. Хранимый видеоряд состоит из 24 кадров, последний из которых является ключевым, так как в нем хранится информация обо всей последовательности. Извлечение осуществляется с помощью вектора признаков, где каждая компонента содержит информацию, полученную из модуля обнаружения событий.

Специфика современных систем видеонаблюдения

В предыдущем разделе мы рассмотрели основные методы «машинного зрения», которые необходимы для обнаружения объекта и распознавания его действий в контексте системы видеонаблюдения. Надо подчеркнуть, что наличие методики или набора методик является необходимым, но недостаточным условием для развертывания крупной системы видеонаблюдения, которая подразумевает сети из множества камер и распределение мощностей обработки сигналов от них.

Методы обработки изображений, используемые в данных системах, основаны на распределении вычислительных мощностей по сети и использовании встроенных в устройства средств обработки, что позволяет повысить производительность и масштабируемость системы.

Подобные пространственно-распределенные среды с мульти-датчиками в настоящее время представляют особый интерес. В последнее время были разработаны специальные методы объединения данных с

различных датчиков [53]. Сложности коммуникации и взаимодействия между разными частями системы играют важную роль в связи с наличием ограничений пропускной способности систем и асимметричности процесса передачи данных [75].

Другим не менее важным аспектом является обеспечение связи между модулями. Кроме того, для некоторых систем наблюдения, возникает необходимость передачи данных через открытые сети, что приводит к появлению задач обеспечения конфиденциальности и аутентификации [5]. В требования к подобным системам также включают возможность автоматического обучения, для ускорения процесса распознавания опасных событий на основе характеристических моделей наблюдаемых объектов [5,6].

Примеры систем видеонаблюдения

Типичными примерами коммерческих систем наблюдения являются DETEC, и Gotcha [27]. В их основе лежат так называемые датчики движения, они могут сохранять в цифровом виде информацию об обнаруженных событиях (ключевых изображениях и метаданные с временными метками), которые, как правило, вызваны появлением новых объектов в сцене. DETEC основана на специальном оборудовании, которое позволяет подключить до 12 камер к одной рабочей станции. Рабочие станции могут быть подключены к сети и все данные наблюдения могут быть сохранены в центральной базе данных, доступной для всех рабочих различных станций в сети. Визуализация исходных изображений с камеры через Интернет подробно описана в статье [30].

Еще одним примером коммерческой системы, предназначенной для применения вне зданий, является DETER (Detection of Events for Threat Evaluation and Recognition). Её задачей является поиск необычных действий среди пешеходов и автомобилей в различных местах, например на

парковках. Система состоит из двух частей: модуль компьютерного зрения и модуль оценки уровня угрозы. Первая часть занимается обнаружением, распознаванием и отслеживанием объектов, для этого система объединяет изображения со всех камер, а затем выполняет слежение за объектами. Оценка угрозы базируется на использовании высокоуровневых семантических методов, оффлайн обучения и онлайнового классификатора возможных угроз. При работе на реальном объекте система показала хорошую производительность в распознавании объектов, однако, как было указано в работе [63], DETER использует относительно небольшое число камер, в силу их высокой стоимости.

Далее можно рассмотреть систему, которая была разработана для итальянских скоростных магистралей [65]. Она состоит из двух основных контрольных комнат, расположенных в удаленных друг от друга местах, а также девяти периферийных контрольных комнат, которые напрямую контролируют всю операционную деятельность: сбор оплаты за проезд, контроль трафика и обслуживание дорожной системы. Большинство сенсоров, используемых для контроля трафика, относятся к классу CCTV, изображения с камер собираются в едином хранилище и могут быть отображены в любой из периферийных контрольных комнат. Камеры, установленные в сложных для контроля местах, относятся к классу управляемых (PTZ). Система самостоятельно способна привлекать внимание оператора к автоматически определенным событиям. Каждая периферийная комната получает свой поток видеоданных в формате MPEG-2 при этом исходящий поток трафика из каждой периферийной комнаты может составлять до 2 Мбит/секунду. Часть периферийных комнат, отвечающих за контроль транспорта в туннелях, оборудованы специальной подсистемой, определяющей остановившиеся в туннеле машины. Основная идея, лежащая в основе этой системы, – иерархичность, что является критически важным параметром для крупных систем видеонаблюдения, поскольку для

предварительной обработки и распознавания данных надежнее и эффективнее использовать модульные решения и не переносить всю вычислительную нагрузку на один сервер.

Авторами в работе [62] представлена система видеонаблюдения, измеряющая параметры трафика. Система осуществляет захват видео с камер, которые размещены на столбах или других объектах. После того как видео оцифровывается и обрабатывается встроенной в камеры аппаратурой, оно передается в единый центр контроля, где производится подсчет статистики, например – среднее время объекта в пути. Вместо использования 3D-моделей транспортных средств авторы отслеживают машины с помощью метрик, например углов входа и выхода машины в область слежения, которые задаются пользователем автономно. С их помощью машины можно объединять и отслеживать в рамках группы. Для объединения в группы система использует графы связности, которые постепенно наполняются углами, соответствующими вершинам автомобилей. К одному компьютеру в такой системе подключаются 17 DSP процессоров (Digital Signal Processors) шесть из тринадцати занимаются отслеживанием, шесть из них отслеживает объекты, четыре – выделяют углы объектов, а один является контроллером слежения, что обеспечивает хорошую производительность системы даже в условиях повышенного трафика.

Авторы работы [33] создали систему, основной целью обнаружения которой являются пешеходы и велосипедисты, которая также способна отслеживать потоки трафика. Система состоит из двух распределенных модулей обработки: модуля слежения, работающего в режиме реального времени и расположенного на столбе рядом с областью наблюдения, и модуля обработки, который работает в режиме оффлайн отдельно. Модуль слежения в данной системе выполняет следующие функции:

- Обнаружение движения;
- Фильтрация изображений;
- Выделение признаков с использованием квази-топологических методов (QTC);
- Отслеживание объектов с использованием фильтров Калмана первого порядка.

Форма и траектория распознанных объектов извлекается и сохраняется в съемной карте памяти, которая потом обрабатывается методом LVQ (Learning Vector Quantisation). У этой системы имеются определенные недостатки. Во-первых, алгоритмы распознавания недостаточно надежны (модель фона не всегда справляется с изменениями и тенями) и зависят от позиции камеры. Во-вторых, несмотря на то, что слежение проводится в режиме онлайн, основной анализ делается в оффлайн режиме и поэтому не представляется возможным вести статистику в режиме реального времени.

Для того чтобы создать систему наблюдения, способную распознавать и обучаться на реальных объектах, очень важно иметь интерпретацию деятельности распознанных объектов. Например, в работе [31], авторы описывают систему видеонаблюдения за парковкой, которая создаёт семантическую интерпретацию распознанных событий и взаимодействий. Такая система состоит из трех частей: устройства слежения, которое отслеживает объекты и разбивает их движения на отдельные части, генератора событий, который создает из них события согласно модели окружения и анализатора, который анализирует события в соответствии со стохастической контекстно-свободной грамматикой (SCFG), описывающей возможные типы деятельности. Следует подчеркнуть, что эта система, как и в работе [51], направлена больше на проверку алгоритмов, чем на создание системы наблюдения на большой территории (система использует всего одну неподвижную камеру). Кроме того, пока неясно, как система может

различать машины и пешеходов, потому что авторы не используют никакой модели формы. Данную проблему мог бы решить модуль распознавания текстовых меток, использование которого позволило бы значительно упростить систему и выделять машину из потока трафика [54].

Уведомление о тревогах и будущее систем видеонаблюдения

Отдельного рассмотрения требуют способы уведомления о тревогах. Например, в статье [32] авторы приводят пример робота, который способен сосредоточить свое внимание на определенной области и динамически переключаться между разными областями. Другой пример приведен в работе [33], где в спецификации системы допускаются конструкции уровня: «набрать номер службы экстренной помощи автоматически, если был обнаружен определенный сигнал». Чтобы иметь возможность выполнять такого рода действия, в систему видеонаблюдения должны быть включены специальные системы контроля и управления.

Можно сделать вывод, что в будущем, создание системы видеонаблюдения за большой территорией будет делаться с использованием разных дисциплин, таких как компьютерное зрение, телекоммуникации и инженерия, так работы [57] посвящены именно проблемам объединения и синергии этих дисциплин. Например, многое могло быть заимствовано из области автономных роботизированных систем, где автономные нецентрализованные агенты взаимодействуют друг с другом в динамической среде. Так как в системах видеонаблюдения много ресурсов тратится на сенсоры и обеспечение надёжности, то такой мультиагентный подход может дать целый ряд преимуществ. Во-первых, объединение агентов позволяет использовать менее дорогие датчики. Во-вторых, увеличивается надежность системы, так как даже если какие-то агенты ломаются, другие могут взять на себя часть их функций. В-третьих,

можно динамически менять нагрузку, распределяя её между группами агентов, что повышает вероятность правильной классификации объекта или цели.

Распознавание автомобильных номеров

Поскольку разрабатываемая система предполагает выявление автомобилей и их идентификацию, наиболее разумным видится способ достижения этого с помощью распознавания автомобильных номеров (текстовых меток).

Важной частью создания систем распознавания текстовых меток является разработка их интеллектуальной составляющей – алгоритмов распознавания текстовых меток. В литературе подробно описаны методы, используемые для решения схожей задачи: распознавания текста. Например, в [28] описан алгоритм функционирования системы FineReader, а работы [47,52,60] посвящены различным аспектам создания автоматизированных систем ввода документов в ЭВМ. В работе [9,10] описаны алгоритмы распознавания рукописных меток. В работе [13] дано описание прикладных систем распознавания таких меток.

Алгоритмам систем распознавания текстовых меток также посвящено достаточно большое количество публикаций. Например, в работе [60] представлены результаты применения различных признаков при построении нейросетевых классификаторов изображений символов идентификационных номеров железнодорожного транспорта.

Реализация аппаратной составляющей систем распознавания текстовых меток связана с решением следующих задач:

- организация ввода видеоряда;
- подбор видеокамеры и объектива с необходимыми характеристиками;
- организация освещения зоны распознавания;
- настройка видеокамеры и пр.

Хотелось бы отдельно отметить достижения теории обработки и анализа изображений, среди которых можно выделить работы как российских, так и зарубежных ученых: Ю.И. Журавлева, В.А. Сойфера, Н.Г. Загоруйко, R. Gonzales, L. Shapiro, M. Haralick, R.O. Duda, P.E. Hart и многих других. Многие из технологий, изначально носящих академический характер, впоследствии находят применение в конкретных (и коммерческих) задачах распознавания текстовых меток.

Многие аспекты разработки систем распознавания текстовых меток уже были исследованы, однако на текущий момент остается ряд нерешенных проблем, которые затрудняют разработку высокоэффективных систем.

Во-первых, это сложность задачи выбора признаков для алгоритмов классификации и построения решающего правила, которые бы позволяли с достаточной надежностью и за приемлемое время классифицировать метки.

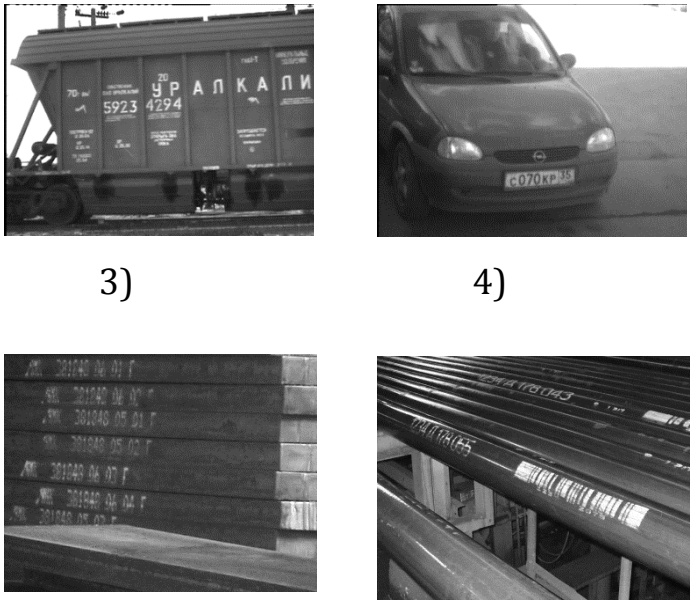
Во-вторых, это проблема формализации задач и понятий, возникающих при построении алгоритмов распознавания текстовых меток. Многие понятия в данной предметной области плохо формализуемы (например «образ символа»), в силу трудности создания точного формального описания эквивалентному понятию человека [19].

В-третьих, это большие сложности создания обучающих выборок с символами и изображениями, которые являются необходимыми как для создания алгоритмов, так и для их внедрения.

Вторая и третья группы проблем обусловлены тем, что системы распознавания текстовых меток постепенно превращаются из штучных лабораторных разработок в промышленные изделия. При таком масштабном тиражировании систем распознавания текстовых меток все чаще составляется из уже готовых программных компонентов, а при внедрении производится их инженерная интеграция.

Системы распознавания используются в составе автоматизированных систем контроля и управления различными процессами, например в промышленности такие системы используются в целях контроля перемещения помеченных изделий, а также в целях автоматизации сбора и сохранения информации об изделиях. Примеры объектов можно увидеть на рисунке ниже:

1) 2)



3) 4)

Рисунок 1.2 Примеры объектов, имеющих метки: 1 – железнодорожные вагоны; 2 – автомобили; 3 – стальные слябы; 4 –трубы

Эти же системы можно использовать для решения вопросов, связанных с контролем передвижения наземных транспортных средств [32]: через КПП, посты полиции, системы СКУД и т.д.

Данные задачи решаются с использованием технологий обработки и анализа кадров видео. В целях дальнейшей работы под видеоизображением мы будем понимать прямоугольное растровое изображение $I_{m\times n}$. Как правило, система производят анализ полутоновых изображений, получаемых посредством видеокамер.

На каждый из объектов наносится маркер: последовательность текстовых символов длинной N символов заранее известного алфавита. Формально маркеру соответствует:

$$C = \{c_1, c_2, ..., c_N\}, \quad (1.1)$$

где $c_i \in \{char_1, char_2, ..., char_S\}$ – алфавит;

S – количество символов алфавита.

В среднем длинна маркера постоянна и в большинстве случаев не превышает двадцати символов алфавита. По сути, текстовая метка является физическим объектом, но формально ей сопоставляется графический образ. Из рисунка 1.2 видно, что обычно системы распознавания оперируют с кадрами, на которых метка занимает сравнительно малый участок от общего изображении. Границы маркера на кадре определяются координатами образов его символов. Однако при создании систем распознавания текстовых меток возникают следующие особенности:

– неравномерная, избыточная или недостаточная освещенность объектов контроля и текстовых меток в зоне контроля;

– непостоянство условий использования (например, при использовании в уличных условиях);

– неопределенная скорость и неизвестный характер движения объектов контроля;

– неподходящие размеры текстовой метки на изображении;

– низкое качество самих меток;

– большие углы поворота и наклона видеокамер по отношению к меткам.

Таким образом, распознавание текстовых меток в видеопотоке сопряжено с рядом специфических проблем, что приводит к необходимости выделить их в отдельный подкласс систем.

Модуль распознавания обычно представляется в виде следующей последовательности алгоритмов:

- Алгоритм локализации,
- Алгоритм сегментации,
- Алгоритм распознавания
- Алгоритм формирования решений
- Алгоритм принятия решений.

Данная схема обусловлена следующими причинами: высокой сложностью распознавания маркера ввиду большого количества состояний [33]. И, во-вторых, возможностью разбить процесс распознавания меток на отдельные простые и логически целостные шаги.

Первый алгоритм (алгоритм локализации [28]) определяет границы текстовой метки на видеоизображении. На входе АЛ получает видеоизображение $I_{m\times n}$, которое с определенной вероятностью содержит текстовый маркер. Видеоизображение является «полезным» (информативным), если оно содержит возможный к прочтению образ текстовой метки, и неинформативным в ином случае. Одним из наиболее распространенных и удачных среди подобных алгоритмов считается алгоритм Кэнни [55].

Результатом работы алгоритма локализации являются координаты (y, x, h, w) зоны $Z_{h\times w}$ на исходном изображении (зона является прямоугольником), либо список вариантов таких зон. Сама локализованная зона представляет собой фрагмент исходного видеоизображения, предположительно содержащий образ текстовой метки и ничего более. Зона является информативной, если она содержит корректно локализованное изображение искомой текстовой метки, в ином случае зона неинформативная.

Алгоритм локализации Кэнни состоит из следующих шагов:

1. Сглаживание. Размытие изображения для удаления шума.

2. Поиск градиентов. Границы отмечаются там, где градиент изображения приобретает максимальное значение. При этом градиенты квантуются по 45 градусов

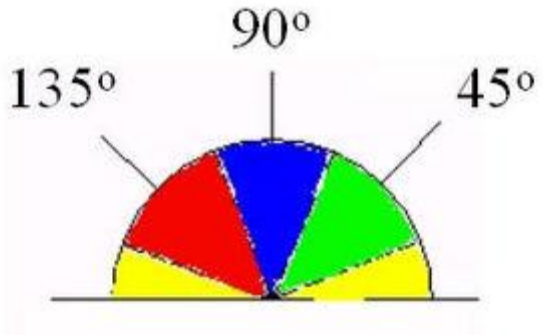


Рисунок 1.3 Направление градиента

3. Подавление не-максимумов. Только локальные максимумы отмечаются как границы.

4. Двойная пороговая фильтрация. Потенциальные границы определяются порогами.

5. Трассировка области неоднозначности. Итоговые границы определяются путём подавления всех краёв, несвязанных с определенными (сильными) границами.

Перед применением детектора, обычно преобразуют изображение в оттенки серого, чтобы уменьшить вычислительные затраты. Этот этап характерен для многих методов обработки изображений.

Следующий алгоритм (алгоритм сегментации) определяет границы символов в выделенном прямоугольнике. Входом данного алгоритма является видеоизображение целиком и координаты прямоугольника внутри него. Выходом данного алгоритма являются координаты отдельных сегментов:

$$S = \{s_1, s_2, \ldots, s_k\}, \quad (1.2)$$

где $s_i = (x_1, y_1, x_2, y_2)$ – координаты i-го сегмента;

k – общее количество выделенных сегментов.

Общее правило таково: Если сегмент является изображением отдельной буквы (символа), то он является полезным («информативным), а в обратном случае – бесполезным («неинформативным»).

Одним из наиболее распространенных и удобных является алгоритм GrabCut. Принцип работы алгоритма GrabCut [1]: Исходя из цветового распределения внутри и снаружи ограничивающего прямоугольника, строится первая цветовая статистика объекта и фона. В качестве цветовой модели используется смесь гауссиан со статически заданным количеством компонент. Затем поочередно производится сегментация (GraphCut-ом, использующим цветовую статистику) и уточнение цветовой статистики. После каждого уточнения цветовой статистики, граф (в котором ищется минимальный разрез) перевзвешивается.

Следующим работает алгоритм распознавания, основанный на нейронной сети, [11]. Данный алгоритм классифицирует изображения отдельных символов. Выход данного алгоритма – это распознанные символы, к которым отнесены входные данные:

$$C = \{c_1, c_2, \ldots, c_k\}, \tag{1.3}$$

где $c_i \in \{char_1, char_2, \ldots, char_n, noise\}$ – алфавит символов меток;

n – общее количество символов в данном алфавите.

Отдельно следует упомянуть класс «шумов». Этот класс состоит из тех изображений не содержащих правильно сегментированного символа. Такие сегменты обычно считаются нераспознанными, а прочие – корректно распознанными. Нейронная сеть, которая используется в системе описана в работе [1].

Итоговое множество, состоящее из распознанных сегментов, анализируется алгоритмом формирования решений. Этот алгоритм производит поиск всех допустимых последовательностей распознанных сегментов, потенциально являющихся образами символов текстовой метки. Результатом работы данного алгоритма является список:

$$M = \{m_1, m_2, \ldots, m_K\}, \quad (1.4)$$

где m_i – код распознанной текстовой метки.

Пример функционирования такой системы распознавания текстовых меток представлен на Рисунке 1.4.

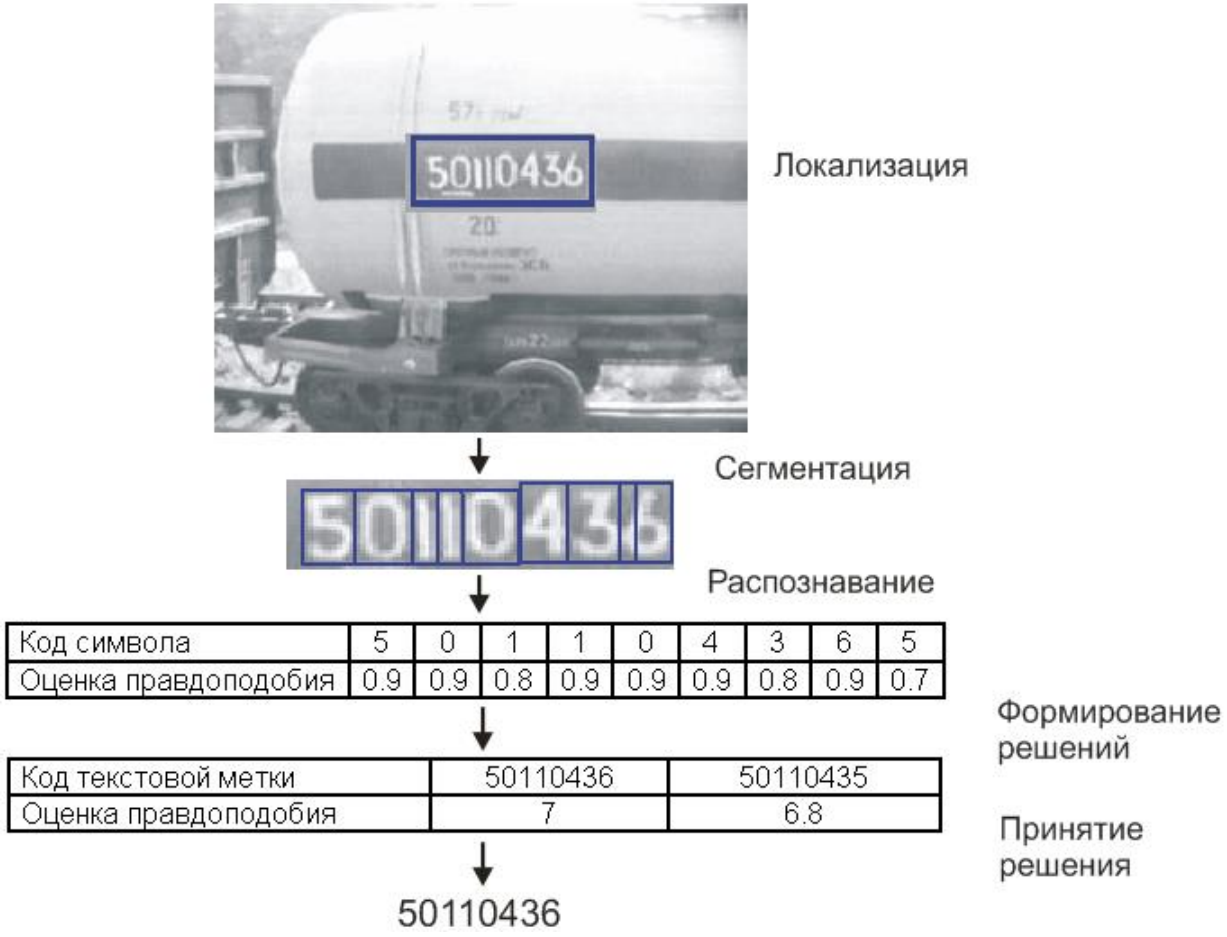


Рисунок 1.4 Этапы анализа видеоизображения

Как отмечено выше, обычно система распознавания текстовых меток функционируют в сложных и нестабильных условиях. Поэтому следующие факторы значительно влияют на работу подобной системы: осадки, посторонние объекты, блики, грязь на метках, низкое качество меток и т.д.

Это приводит к сложностям реализации

Наличие данных проблем приводит к тому, что задачи и понятия, возникающие при разработке алгоритмов система распознавания текстовых меток, являются трудно формализуемыми. Это приводит к применению специальных эвристических алгоритмов в составе таких систем. Такие алгоритмы получили большое распространение после появления и дальнейшего развития теории распознавания образов [14].

Таким образом, основными критериями корректной работы системы распознавания текстовых меток, являются некоторые вероятностные характеристики. Такие как:

- вероятность правильного распознавания метки P_{right},
- вероятность неправильного распознания метки P_{err},
- вероятность ложного распознавания на неинформативном видеоизображении P_{false}.

Оценка значений указанных вероятностных критериев выполняется на основе статистики. Для этого используется обучающая последовательность, которая представляет собой набор видеоизображений и соответствующих им текстовых меток.

Не следует забывать про такую важную характеристику как быстродействие, поскольку обычно система распознавания должна работать в режиме реального времени. Поэтому одним из критериев эффективности всей системы является задержка времени, вызванная временем распознавания отельного кадра.

Сведем воедино все критерии эффективности работы системы:

$$Q = (P_{right}, P_{err}, P_{false}, T) \quad (1.5)$$

Отсюда следует, что эффективность системы распознавания текстовых меток носит неоднозначный характер, при этом критерии входят в противоречие друг с другом: улучшение одного критерия обычно приводит к ухудшению других. Из-за этого применяются компромиссные решения, которые, как правило, максимально улучшают значение P_{right} при ограничениях на все остальные:

$$P_{right} \to \max, \; P_{right} \geq P_{right}^{tr}, \; P_{err} \leq P_{err}^{tr}, \; P_{false} \leq P_{false}^{tr}, \; T \leq T^{tr}. \quad (1.6)$$

Данные значения задаются, исходя из задач, которые должна решать система идентификации. Первой задачей является идентификация всех объектов, которые фиксирует система. Второй задачей является поиск предварительно заданных объектов среди всех зафиксированных системой. В обеих задачах особенно важен P_{right}, однако при решении последней ограничения на значения P_{err} и P_{false} обычно значительно более мягкие.

Выводы по первой главе

1. Большинство современных систем бихевиористического анализа основаны на видеонаблюдении. В целях повышения стабильности такие системы являются распределенными и модульными. Также их можно назвать «облачными» ввиду наличия у многих из них единого координирующего центра, хранящего большую часть данных системы.

2. Большинство систем реализуется на стандартном и легко заменяемом оборудовании, что повышает их надежность и стабильность работы, что крайне важно для системы обеспечения безопасности, которая должна работать в режиме реального времени.

3. Системы бихевиористического анализа могут работать в Real-time режиме, не требуя значительных вычислительных мощностей – достаточно относительно современных настольных компьютеров со стандартным системным программным обеспечением.

4. Возможности алгоритмов распознавания изображений и машинного зрения в современных системах используются недостаточно эффективно.

5. Существующие системы выявления нелегитимной активности на основе видеонаблюдения используют большое число камер, громоздкие алгоритмы и показывают низкую эффективность.

Глава 2. Модель системы выявления нелегитимных действий

Контекст решаемой задачи

Основная задача, которую должны решать используемые в системе критерии – выявлять нелегитимную активность (попытки незаконного вывоза продукции), путем анализа данных об автомобилях, проезжающих через промышленную площадку. Исходя из этой постановки, опишем кратко контекст задачи выявления нелегитимных действий.

- Объект – промышленный объект, производящий различную продукцию. У него следующие характеристики:
  - Огороженная территория.
  - Ввоз-вывоз продукции производится только через КПП.
  - Ввоз-вывоз продукции производится на большегрузном транспорте.
  - Территорию объекта можно представить в виде связного графа, где вершины инцидентности один – КПП, инцидентности больше единицы – цеха и места разгрузки-погрузки продукции.
- КПП – точка ввоза-вывоза продукции с объекта.
  - Каждый КПП оснащен двумя направлениями – одним на выезд, другим – на въезд.
  - На каждое из направлений (въезд и выезд) существует заградительный элемент – шлагбаум, управляемый дистанционно.
- Событие – проезд автотранспорта через КПП объекта.
- Нелегитимное событие – событие, при котором проезд автотранспорта был произведен незаконно.

- Технически возможно оснастить:
    - Системой видеонаблюдения – КПП.
    - Датчиками обнаружения металлических объектов (автомобилей) – КПП и цеха.
    - Вычислителями – КПП.
    - Датчиками открытия-закрытия шлагбаумов – КПП.
    - Объединить всю систему с помощью ЛВС с ограниченной пропускной способностью.
- Признаки – дискретные значения
    - Вычисляются на основе данных, поступивших на вычислители по заданным алгоритмам, в том числе и из системы видеонаблюдения.
- Задача: необходимо разработать систему, которая выявляет нелегитимные события на основе набора признаков нелегитимной активности.

Выделение признаков нелегитимной активности

Наиболее сложной задачей является выявление искомого набора признаков нелегитимной активности. Каждый из них должен обладать следующими двумя характеристиками:

- Вычислимость, т.е. то, что признак возможно вычислить на основе данных поступающих в систему.
- Релевантность, т.е. то, что он характеризует событие, как имеющее отношение к легитимному, так и нет в достаточной степени.

Если первая характеристика не вызывает затруднений, то вторая относится к области поведенческого анализа людей и выявления преступников среди обычных граждан.

В настоящее время для этого используются различные подходы, наиболее технологичным из которых является анализ выражения лиц людей – у преступников оно отличается от обычных граждан по ряду ключевых характеристик, которые возможно распознать с помощью системы распознавания выражений человеческих лиц [48]. Однако, такие системы обладают определенными недостатками, в частности:

- Алгоритмы распознавания лиц, несмотря на все достижения машинного зрения, допускают многочисленные ошибки, как первого, так и второго рода.
- Для распознавания выражения лиц в режиме реального времени необходимы значительные вычислительные мощности [48].
- Невербальные проявления потенциального преступника еще недостаточно изучены и вызывают многочисленные споры, как среди психологов, так и криминалистов.

Однако, достаточно давно был разработан метод выявления преступников на основе их поведенческого анализа, которым пользуются службы в аэропортах всего мира, в частности служба Transportation Security Administration в США. Эти службы используют достаточно простые и в тоже время эффективные и проверенные признаки того, что пассажир может быть потенциально опасным [54]. Эта идея была предложена и впервые реализована в аэропортах Израиля, где на текущий момент, во многом благодаря этому методу, выявляется более 90% процента потенциальных преступников и террористов.

Этот метод можно назвать *анализом шаблонов преступного поведения.* Такие шаблоны применяются не только в аэропортах, но и охранниками в бизнес-центрах, режимных объектах и т.д.

В данной работе автором предложена следующая идея *–досмотр пассажира по своей сути ничем не отличается от досмотра автомобиля на выезде с КПП, поэтому принципы поведенческого выявления потенциальных преступников допустимо применять к автотранспорту на КПП.*

Если отталкиваться от этой идеи, то допустимо на основе шаблонов нелегитимной активности разработать критерии выявления потенциальных преступников. Однако, даже создатели этой программы отмечают, что помимо разработки самих критериев критически важным является наличие персонала, способного замечать подобное поведение, притом последнее они считают даже более значимым, чем разработку критериев. Поэтому система, автоматически распознающая подобные признаки, должна выявлять их с максимальной достоверностью. Именно поэтому признаки, выбранные автором, максимально прозрачны и устойчивы к потенциальным ошибкам в их вычислении.

Также следует отметить, что поскольку разрабатываемая система должна быть масштабируемой и адаптируемой, то необходимо реализовать настраиваемый программный модуль, позволяющий гибко менять, используемые в системе признаки. Описание такого модуля приводится в третьей главе.

Шаблонов определения нелегитимной активности достаточно много и все их приводить не имеет смысла (список программы TSA содержит более трех тысяч пунктов [54]), тем более что некоторые признаки достаточно очевидны и без шаблонов, приведем лишь те, которые в результате были использованы в разрабатываемой системе [1,2]:

- Нетипичное время осмотра охранником на входе в здание, при наличии металлодетектора. В 79% случаев аномально долгий осмотр охранником означает нежелательного гостя. В случае разрабатываемой системы была также выдвинута гипотеза о

подозрительности очень быстрого осмотра автомобиля – все машины большегрузные и для каждой из них осмотр должен быть максимально тщательным, а быстрый осмотр не представляется возможным.

- Проход в здание в непредусмотренное время. В 63% случаев проход в здание в нерабочее время (особенно глубокой ночью) означает нелегитимную активность. Еще более высокая вероятность в случае прохода на международный рейс более чем за 7 часов до его вылета (82%) [54].
- Резкие изменения внешнего вида. Темные очки в 71% случаев означают попытку скрыть свою личность и пройти по чужому пропуску или паспорту.

Несмотря на некоторую простоту, эти признаки играют огромную роль в выявлении потенциальных террористов – основная задача не создать новые сложные признаки, а эффективно применять существующие [54]. До текущего момента такой подход ни разу не применялся к автотранспорту промышленных объектов и является новым для систем безопасности.

На основе вышеперечисленных шаблонов были разработаны признаки, которые в дальнейшем тексте идут под номерами один, три и восемь. Остальные пять были разработаны на основании анализа зарегистрированных случаев попыток краж на предприятиях и их смысл достаточно очевиден. Формализованное описание признаков приводится в следующем разделе.

Формальная постановка задачи

Пусть $x_1, \dots, x_n$ – различные признаки, каждый из которых принимает конечные дискретные значения. И пусть S – множество событий. При этом каждому событию $s \in S$можно однозначно сопоставить вектор $\bar{x} = (x_1, \dots, x_n)$из признаков.

Множество Sсостоит из двух подмножеств: легитимные события L и нелегитимные N. При этом:

$$S = L \cup N, L \cap N = \emptyset.$$

Назначим легитимным событиям - 1, а нелегитимным – 0. Также определим обучающую выборку $S' \in S$ и $L' \in L$. В таком случае, по обучающей выборке S' и L'необходимо построить отображение α, которое:

$$\forall \bar{x} = (x_1, \dots, x_n) \in S,\ \alpha: \bar{x} \rightarrow \{0,1\} \textit{ (2.1)}$$

Таким образом, система должна решать задачу классификации, то есть по некоторой конечной обучающей выборке относить событие по ряду признаков или к легитимным или к нелегитимным событиям.

Описание модели системы

В данном разделе будет описана модель выявления нелегитимных действий, лежащая в основе разработанной системы. Отметим, что для каждого используемого критерия приводится его описание на псевдокоде, используемом в системе для описания и настройки различных критериев. Подробное описание программной части реализации критериев приводится в третьей главе.

Целевая функция

Исходя из формальной постановки задачи, целевая функция αимеет вид:

$$\forall \bar{x} = (x_1, \dots, x_n) \in S,\ \alpha: \bar{x} \rightarrow \{0,1\}.$$

Она относит событие с признаками $(x_1, \dots, x_n)$ к классу легитимных – 1 или нелегитимных – 0 событий.

Входные данные

Входными данными в решаемой задаче являются дискретные признаки $x_1, \dots, x_n$. Фактическое значение n = 8. Перечислим все

используемые признаки с их кратким описанием. Признаки были выбраны эмпирическим путем и исходя из технических ограничений на получаемые данные.

1. Мониторинг времени прохождения осмотра автомобилем

Интерпретация признака: если автомобиль на выезде стоит на осмотре дольше некоторого критического времени, то вероятность того, что произошло нарушение, растет пропорционально времени осмотра. Аналогичная логика применяется и для очень маленького времени осмотра.

Единица измерения: секунды

Эталонные значения: вычисляется как среднее арифметическое всех времен досмотра тестовой выборки, т.н. $t_{\text{среднее}}$ – среднее время осмотра.

Формула подсчета $x_1 = mod\left(t_{\text{среднее}} - t_{\text{фактическое}}\right)$ где: $t_{\text{фактическое}}$ – фактическое время осмотра автомобиля в секундах.

Принимаемые значения:$x_1 \in \overline{1, \dots, 7200} \in N$

Описание на псевдокоде:

time() – функция возвращающая текущее системное время.

t_mean – среднее время осмотра, предварительно определенная константа.

event(описание события) – функция, ожидающая наступления в системе определенного события.

alert(); функция оповещения.

event(carInSight);

t_1:=time();

event(barrierIsOpen);

t_2:=time();

x_1 := mod(t_mean – (t_2-t_1));

2. Мониторинг времени движения между различными точками контроля

Интерпретация признака: измерение времени движения автомобиля из одной точки маршрута в другую. Если автомобиль двигался нетипично долго – есть подозрение, что с грузом были совершены противоправные действия. Статистически считаем время движения объекта.

Единица измерения: секунды

Эталонные значения: $t_{\text{среднее}}$ – среднее время движения автомобиля по его маршруту движения, Δ – допустимое отклонение:

Формула подсчета: $x_2 = mod\ (\Delta - t_{\text{фактическое}} - t_{\text{среднее}})$, где: $t_{\text{фактическое}}$ – фактическое время движения по маршруту в секундах.

Принимаемые значения:$x_2 \in \overline{0, \dots, 4000} \in N$

Описание на псевдокоде:

estimatedTravelTime(); функция, возвращающая время движения автомобиля, согласно транспортным документам – реализована системой электронного документооборота

event(carInSight);

t_enter:= time();

event(carInSightOtherControlPoint);

t_exit:=time();

x_2 := mod(delta – t_enter – t_exit);

3. Контроль проезда машины в непредусмотренное время

Интерпретация признака: каждая машина однозначно идентифицируется по номеру, а также по заявке на въезд, в которой написано ее предполагаемое время прибытия на территорию объекта. Если машина въезжает на территорию в неустановленное время существует вероятность того, что данный въезд не является легитимным.

Единица измерения: секунды

Эталонные значения: $t_{\text{предполагаемого проезда}}$ – предполагаемое время проезда, $t_{\text{фактического проезда}}$ – фактическое время проезда. Оба измеряются в секундах от 0:00:00 дня предполагаемого въезда.

Формула подсчета: $x_3 = mod\ (t_{\text{предполагаемого проезда}} - t_{\text{фактического проезда}})$

Принимаемые значения: $x_3 \in \overline{0, \dots, 180000} \in N$.

Описание на псевдокоде:

enterTime(); функция, возвращающая время въезда автомобиля, согласно транспортным документам – реализована системой электронного документооборота

event(carInSight);

t_enter:= time();

x_3:= mod(enterTime() – t_enter);

4. Принятие решения на возможность проезда непредусмотренным лицом

Интерпретация признака: у каждого охранника есть своя именная карточка, путем поднесения которой к считывателю дается разрешение машины на въезд и производится открытие шлагбаума. На объекте ведется график смен охранников. Поэтому если открытие было произведено охранником не из текущей смены, это также является признаком

нелегитимного действия. Система может сама выдавать значения – 1 в случае охранника текущей смены и 0 – в обратном случае.

Принимаемые значения:$x_4 = \{0,1\} \in N$

Описание на псевдокоде:

allowedList(); функция, возвращающая допустимый список охранников в текущую смену – реализована системой электронного документооборота

currentOpener(); –функция возвращающая охранника открывшего шлагбаум

x_4:=0;

event(barrierIsOpen);

if currentOpener() not in allowedList() x_4:=1;

5. Изменение картины воздействия на датчики магнитного поля

Интерпретация признака: На каждом направлении движения на КПП установлены датчики, измеряющие степень возмущения магнитного поля земли (это позволяет определить факт подъезда машины к шлагбауму). Для каждой машины всегда известны две характеристики: предполагаемое возмущение (исходя из типа машины и груза), а также фактическое возмущение. Оба этих значения измеряются в микротеслах.

Единица измерения: микротеслы.

Эталонные значения: в μT

Формула подсчета: $x_5 = mod\ (T_{\text{предполагаемое}} - T_{\text{фактическое}})$

Принимаемые значения:$x_5 \in \overline{0, \dots, 1000} \in N$

Описание на псевдокоде:

proposedTesla(); функция, возвращающая допустимое возмущение магнитного поля исходя из типа груза – реализована системой электронного документооборота

currentTesla(); –функция возвращающая охранника открывшего шлагбаум

event(barrierIsOpen);

x_5:= mod(proposedTesla() – currentTesla());

6. Контроль проезда дополнительных точек

Интерпретация признака: при въезде автомобиля на территорию объекта водителю в пропуск выдается специальная метка дальнего действия. На территории объекта во всех местах вероятной погрузки и выгрузки установлены считыватели данных меток. Это позволяет определить места остановки водителя. С учетом того, что системе (благодаря данным из товарной накладной) известен маршрут движения автомобиля, возможно оценить в каком количестве неустановленных мест останавливался водитель.

Принимаемые значения:$x_6 \in \overline{0, \dots, 100} \in$

Описание на псевдокоде:

pointsWay(); функция, возвращающая количество точек, которое машина должна проехать – реализована системой электронного документооборота

pointsTraveled(); функция, возвращающая количество точек, которое фактически проехала машина

event(CarInSight);

x_6 := pointsWay() – pointsTraveled();

7. *Контроль непрохождения обязательных точек*

Интерпретация признака: аналогично предыдущему признаку, только в данном случае оценивается какое количество предполагаемых точек движения не проехал автомобиль.

Принимаемые значения: $x_7 \in \overline{0, \dots, 100} \in N$

Описание на псевдокоде:

pointsWay(); функция, возвращающая количество обязательных точек, которое машина должна проехать – реализована системой электронного документооборота

pointsTraveled(); функция, возвращающая количество обязательныхточек, которое фактически проехала машина

event(CarInSight);

x_7 := pointsWay() – pointsTraveled();

8. *Фиксация критической разности качества распознавания номера*

Интерпретация признака: водитель может попытаться обмануть систему путем скрытия номера средства: замазывания грязью и т.д. Система распознавания номеров может оценить, насколько хорошо был распознан номер по шкале от одного до десяти. Разность значений, измеренная на двух разных КПП, может являться одним из критериев выявления нелегитимных действий.

Единица измерения: натуральные числа.

Эталонные значения: отсутствуют.

Формула подсчета: $x_8 = mod\ (Q_{\text{въезда}} - Q_{\text{выезда}})$, где $Q_{\text{въезда}}$ – качество распознавания, измеренное на въезде, где $Q_{\text{выезда}}$ – качество распознавания, измеренное на выезде.

Принимаемые значения:$x_8 \in \overline{0,\dots,10} \in N$

Описание на псевдокоде:

recognitionQuality(); функция, возвращающая качество распознавания,

event(CarInSight);

temp:=recognitionQuality();

event(CarExit);

x_8:=mod(recognitionQuality() –temp);

Выбор алгоритма классификации

Следующим возникает вопрос: какой алгоритм классификации использовать для выделения легитимных и нелегитимных действий. Поскольку существует множество алгоритмов классификации объектов, приведем их краткое описание и выделим основные достоинства и недостатки.

- Построение дерева решений.
    - В процессе построения модели, алгоритм вычисляет степень влияния и использует входной параметр, который оказывает наибольшее влияние на значение выходного атрибута, для разбиения узла дерева решений.
    - Достоинства: в общем случае обеспечивает высокую скорость работы.
    - Недостатки: сложная структура. Для построения дерева решений или для внесения изменений, требуется помощь специалиста в рассматриваемой области.
    - Время работы алгоритма:

$$T(n) = O\left(n - \frac{n}{n-p}\right)$$

где n — количество условий, p — среднее количество переходов.

- Метод Байеса
  - Для каждого наблюдения из классифицируемого множества высчитывается вероятность его возникновения в соответствующем классе по формуле условной вероятности Байеса.
  - Достоинства: высокая скорость работы алгоритма как на этапе обучения, так и на этапе анализа новых данных
  - Недостатки: перемножать условные вероятности корректно только тогда, когда все входные переменные действительно статистически независимы.
  - Время работы алгоритма:

$$T(n) = O(nk),$$

где n — размер множества X , k — размер множества Y (X – множество классифицируемых объектов, Y–множество классов).

- Алгоритм SVM
  - В основе алгоритма лежит идея разделения облаков данных гиперплоскостями, находящихся на максимальном расстоянии от облаков.
  - Достоинства: применение гиперплоскостей работает при малых объемах обучающей выборки, за счет использования ядра описывающего связь между элементами выборки, можно использовать гиперплоскости разной сложности.
  - Недостатки: большое время на обучение системы.

- Время работы алгоритма:

$$T(n) = O(n^3)$$

где n — размер множества X.

- Метод k-средних
  - Метод основан на разделении множества наблюдений X на k кластеров, которые локально минимизированы относительно расстояния между информационной точкой и центроидом кластера.
  - Достоинства: можно выделить кластер для данных после нахождения центроидов.
  - Недостатки: субъективность –результат работы зависит от критериев, выбранных специалистом.
  - Время работы алгоритма

$$T(n) = O(n^{kd})$$

где n — размер множества X , k — количество кластеров, d — размерность X.

- Иерархическая кластеризация
  - Основная цель алгоритма заключается в построении структуры кластеров. Можно выделить два вида алгоритма:
    - Объединяющий.
    - Разделяющий.

Расстояние между кластерами принимают за меру, используемую для определения, какие кластеры должны быть объединены, а какие кластеры требуется разделить.

- Достоинства: высокая скорость работы, простота реализации, возможность просмотреть результаты работы на каждом ходу.

- Недостатки: нельзя напрямую управлять количеством кластеров. Субъективность –результат зависит от критериев выбранных специалистом.
- Время работы алгоритма

$$T(n) = O(n^2)$$

где n — размер множества X .

- EM-кластеризация(Expectation-Maximization)
  - В рамках данного алгоритма вместо центров кластеров предполагается наличие функции плотности вероятности для каждого кластера с соответствующим значением математического ожидания и дисперсией. Перед стартом алгоритма выдвигается гипотеза о виде распределений, которые сложно оценить в общей совокупности данных.
  - Достоинства: не требует выделения специалистом метрик, есть возможность использовать совместно с другими алгоритмами обработки данных. Может работать на малых объемах данных.
  - Недостатки: результат работы зависит от первоначального вида распределения.
  - Время работы алгоритма

$$T(n) = O(n^2)$$

где n — размер множества X.

Опишем ограничения и условия, накладываемые на выбранный алгоритм в рамках данной системы:

- Алгоритм классификации перенастраивается раз в сутки с учетом вновь поступивших данных.
- Время обучения алгоритма не должно превышать 10-15 минут.

- Если за сутки не было ошибок в классификации (определяется оператором раз в сутки), то обучение не производится.
- Обучения производится по легитимной выборке (подготовленной человеком).
- Порядок размера множества$X - 10^6$.

Выбор алгоритма производился эмпирическим путем на выборке из 2 450 567 легитимных выездов. Оценивалось количество ошибок первого и второго рода. Результаты приведены в таблице ниже.

Таблица 2.1 Сравнение алгоритмов классификации

Алгоритм	Ошибки первого рода	Ошибки второго рода
Дерево решений (алгоритм C4.5)	28	14
Метод Байеса	39	56
SVM	34	51
k-средних	28	113
Иерархическая кластеризация	13	18
EM-кластеризация	10	7

Из таблицы видно, что наименьшую ошибку дает алгоритм EM-кластеризации, а наихудший результат показал SVM. Время вычисления EM на машине с процессором Core i5 2700 составило 10 секунд, при реализации алгоритма на платформе .Net, что допустимо в рамках данной системы. По результатам исследования было принято решение использовать EM-кластеризацию в качестве основного алгоритма формируемой системы видеонаблюдения.

Выбор и оценка эффективности алгоритма распознавания номеров

Следующим шагом при построении системы должен быть реализован программный модуль распознавания номеров. Однако предварительно, необходимо описать алгоритмическую часть данного модуля (большая часть была описана в первой главе). По результатам обзора были выбраны алгоритмы сегментации, локализации, распознавания и принятия решений, все из которых являются стандартными и реализованными в других работах. С точки зрения разработки системы основной вопрос – какой процент проезжающих автомобилей будет распознавать разработанная система. Поэтому далее приводится математическая модель комбинации этих алгоритмов, позволяющая оценить эффективность работы – вероятность правильного распознавания автомобильного номера.

Формула оценки эффективности модуля распознавания номеров

Процесс функционирования алгоритма $A \in A^C$ можно представить в виде последовательности событий, характеризующих корректность результата анализа входных данных алгоритмами-компонентами A_i, $i = 1..n$. Каждая такая последовательность приводит к одному из исходов анализа входных данных. Таким образом, вероятность того или иного исхода анализа входных данных алгоритмом A зависит от вероятностей исходов анализа соответствующих данных алгоритмами-компонентами.

Отсюда следует функциональная зависимость значений критериев эффективности Q от Q^i, $i = 1..n$. Значения критериев эффективности P_{right}, P_{err}, P_{false} вычисляются из значений критериев эффективности $P(C_i)$, $P(F_i)$ следующим образом:

$$P_{right} = P(C_1 \mid I)P(C_2 \mid IC_1)...P(C_n \mid IC_1C_2...C_{n-1}), \quad (2.1)$$

$$P_{err} = 1 - P_{right} - P_{empty}, \quad (2.2)$$

$$P_{empty} = P(\overline{F}_1 \mid I) + P(\overline{F}_2 \mid IF_1) + \ldots + P(\overline{F}_n \mid IF_1F_2\ldots F_{n-1}), \quad (2.3)$$

$$P_{false} = P(F_1 \mid N)P(F_2 \mid NF_1)\ldots P(F_n \mid NF_1F_2\ldots F_{n-1}), \quad (2.4)$$

где P_{empty} – вероятность пустого выхода алгоритма $A \in A^C$ при информативном входе.

События I = «на вход $A \in A^C$ поступили информативные данные» и N = «на вход $A \in A^C$ поступили неинформативные данные» указывают на то, что значения критериев Q рассчитываются отдельно для информативных и неинформативных входов, вероятности этих событий здесь равны единице.

Вероятности событий условны, так как на вход алгоритма поступают только те данные, которые являются выходом вышестоящих алгоритмов. Поэтому экзаменационная последовательность алгоритма A_i формируется композицией вышестоящих алгоритмов. Например, экзаменационную последовательность АЛ составляют непосредственно экзаменационные видеоизображения. Выходы АЛ образуют экзаменационную последовательность для АС. Чтобы оценить значения вероятностных критериев АЛ и АС, требуется указать координаты текстовых меток и координаты символов на экзаменационных изображениях. То есть для расчета условных вероятностей событий C_i, F_i требуется сформировать «общую» экзаменационную последовательность – массив прецедентов [14].

Массив прецедентов представляет собой множество пар данных: входных данных алгоритма и соответствующие эталонные выходные данные всех или некоторых этапов анализа входных данных. Массив прецедентов СРСМ для каждого изображения содержит эталонные выходы: координаты образа текстовой метки, координаты образов символов, коды символов и другую информацию. В простейшем случае массив содержит лишь код текстовой метки для каждого из экзаменационных изображений.

Массив позволяет оценивать значения вероятностных критериев эффективности алгоритмов-компонентов так, что значения оценок характеризуют эти алгоритмы как составную часть системы распознавания символьных меток. Сформировать такой массив прецедентов может практически любой человек, обладающий способностью читать символы метки на видеоизображении и вводить данные в ЭВМ, поэтому для этой работы не требуется привлекать специалистов, обладающих какими-либо специфическими навыками.

Процесс функционирования алгоритма $A \in A^C$ может быть представлен в виде последовательности событий C_i, $\overline{C}_i$, F_i, $\overline{F}_i$, которые наступают в процессе анализа входных данных алгоритмами-компонентами. Совокупность таких последовательностей образует двоичное дерево – дерево событий СРСМ. Произвольная вершина дерева соответствует наступлению (либо не наступлению) одного из указанных событий, а все предки вершины, начиная с корня дерева, соответствуют хронологически предшествующим событиям. Каждая вершина характеризуется совместной вероятностью событий, образующих путь к данной вершине, исходящий из корня. Дерево событий позволяет с учетом декомпозиции A визуально отобразить последовательности событий, приводящие к разным исходам анализа входных данных. На Рисунке 2.1 представлен пример такого дерева, характеризующего систему распознавания, алгоритмами-компонентами которой являются АЛ, АС, алгоритм распознавания зоны (композиция АР и АФР) и АПР.

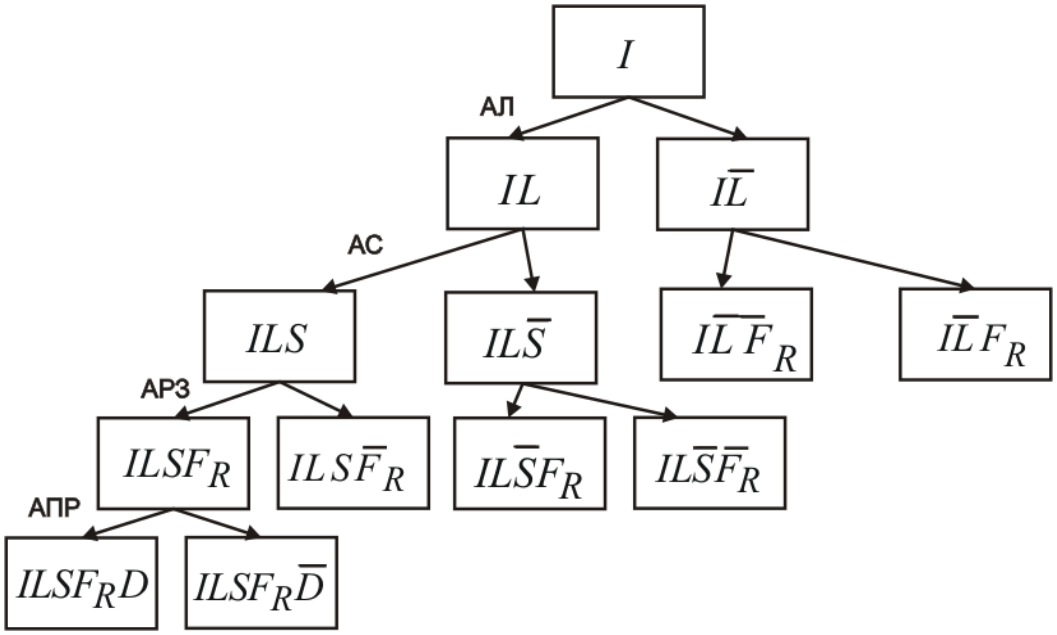


Рисунок 2.1 Пример дерева событий СРСМ

Обозначения событий, которые представлены на рисунке, даны в таблице 2.2.

Таблица 2.2 События, характеризующие исходы анализа алгоритмами СРСМ входных данных

Алгоритм	События	
	Корректность решения	Полнота решения
АЛ	L = «локализована информативная зона»	F_L = «локализована неинформативная зона» (если АЛ может возвращать пустой выход)
АС	S = «образы всех символов корректно сегментированы»	F_S = «выход сегментации не пустой» (если АС может

		формировать пустой выход)
АР	K = «правильно классифицированы образы всех N символов метки»	F_K = «к классу символов отнесено N и более сегментов»
АФР	R = «среди сформированных решений есть правильное»	F_R = «сформировано хотя бы одно решение»
АПР	D = «принято правильное решение»	F_D = «принято решение» (если АПР может выдавать пустой выход)

Предполагается, что выходы АЛ, АС и АПР всегда не пусты. Поэтому дерево событий на рисунке включает лишь одно событие вида «выход алгоритма пуст», характеризующее композицию АФР и АР. Корень дерева соответствует событию I = «на вход системы распознавания поступило информативное видеоизображение», вероятность этого события считается равной 1 и может не использоваться при расчетах. Вершины-листья соответствуют исходам анализа видеоизображения, причем одному исходу может соответствовать несколько вершин-листьев. Дерево на Рисунке 2.1 позволяет установить зависимость значений вероятностных критериев эффективности СРСМ от значений критериев алгоритмов-компонентов:

$$P_{right} = P(L)P(S \mid L)P(F_R \mid LS)P(D \mid LSF_R), \quad (2.5)$$

$$P_{err} = P(L)P(S \mid L)P(F_R \mid LS)P(\overline{D} \mid LSF_R) + \quad (2.6)$$
$$+ P(L)P(\overline{S} \mid L)P(F_R \mid L\overline{S}) + P(\overline{L})P(F_R \mid \overline{L}),$$

$$P_{false} = P(F_R \mid N). \quad (2.7)$$

Следует отметить, что на Рисунке 2.1 представлено дерево событий, которые могут наступить при обработке информативного

видеоизображения. Процесс анализа неинформативных видеоизображений отражается его правым поддеревом.

Таким образом, удалось установить зависимость значений вероятностных критериев эффективности СРСМ от значений критериев алгоритмов-компонентов модуля распознавания номеров.

Оценка вероятности правильного распознавания в зависимости от времени анализа кадра

В большинстве случаев системы распознавания автомобильных номеров функционируют в составе систем идентификации. Идентификация объектов по их текстовым меткам обеспечивает возможность инициации каких-либо управляющих воздействий на появляющиеся в зоне контроля объекты. Поэтому при использовании СРСМ в составе таких систем значимым требованием к алгоритму является заданный режим быстродействия. Показателем этой способности является задержка времени T с момента поступления видеоизображения на вход СРСМ до момента появления результатов распознавания.

Требование к величине задержки определяется техническими условиями эксплуатации СРСМ. Величина задержки T зависит от характера анализируемых видеоизображений, от алгоритмов СРСМ и их параметров, а также от быстродействия ЭВМ. Поэтому быстродействие системы может повышаться как за счет использования высокопроизводительного аппаратного обеспечения, так и за счет совершенствования алгоритмов и программного обеспечения, реализующего алгоритмы. Использование высокопроизводительных ЭВМ увеличивает стоимость оптоэлектронной системы, помимо этого потребители часто заинтересованы в использовании компактных промышленных компьютеров, поэтому именно совершенствование алгоритмического обеспечения СРСМ является актуальной задачей.

Каждый алгоритм-компонент A_i характеризуется средним временем анализа T_i единицы входных данных. На практике может быть найдена некоторая оценка $\hat{T}_i$, поскольку в общем случае время может варьироваться в некоторых пределах. Это обусловлено тем, что, во-первых, время анализа алгоритмом разных входных данных может отличаться, во-вторых, время анализа алгоритмом одинаковых входных данных может отличаться.

Первая причина связана с нестационарностью среды функционирования СРСМ, то есть неопределенностью условий формирования видеоизображений. Вследствие этого данные, передаваемые между алгоритмами, могут изменяться как качественно, так и количественно: например, размеры сегментированных зон в СРСМ могут варьироваться в некоторых рамках, что влияет на время обработки зоны алгоритмом сегментации. Характер изображений на локализованных зонах также изменяется, вследствие чего может увеличиться или уменьшится количество выделенных сегментов и, соответственно, время анализа зоны алгоритмами распознавания и принятия решений. Вторая причина связана с тем, что программное обеспечение СРСМ часто работает под управлением распространенных операционных систем, не являющихся системами реального времени.

Время T во многом определяется вероятностями событий F_i, поскольку, как правило, реализация алгоритма A такова, что он прекращает анализ входных данных при появлении пустого либо неполного (недостаточно сегментов или распознанных символов для формирования решения) выхода одного из алгоритмов-компонентов A_i.

Отсюда при декомпозиции $A \in A^C$ на n алгоритмов-компонентов получаем:

$$T = T_1 + T_2 P(F_1) + T_3 P(F_2 \mid F_1) + \ldots + T_n P(F_n \mid F_1 F_2 \ldots F_{n-1}) \quad (2.8)$$

Установленная зависимость позволяет в процессе разработки СРСМ оценивать значение параметра T без проведения непосредственных измерений, ограничиваясь вычислением характеристик того алгоритма, который подвергся изменениям.

СРСМ обычно функционируют в режиме реального времени, поэтому критерий T является важной характеристикой СРСМ. Время анализа видеоизображения определяет количество кадров, которое способна обрабатывать система за единицу времени, и, следовательно, непосредственно влияет на вероятность идентификации P_{id} объекта контроля.

Время анализа видеоизображения определяет количество кадров, которое способна анализировать система контроля за единицу времени, и, следовательно, непосредственно влияет на вероятность идентификации объекта контроля P_{id}. Рассмотрим зависимость вероятности события Id =«объект идентифицирован» от T .

Вероятность идентификации объекта рассчитывается следующим образом:

$$P(Id) = \sum_{i=1}^{N} P(K_i) \quad (2.9)$$

.

где N – количество кадров, содержащих образ каждого объекта и его текстовой метки. Пусть N одинаково для каждого объекта;

t_O – время наблюдения текстовой метки объекта при его перемещении через зону контроля (длительность присутствия текстовой метки «в кадре»);

$P(K)$ – распределение вероятности количества кадров, на которых метка корректно распознана;

$P(K_i)$ – вероятность того, что метка распознана ровно на i кадрах из N.

Плата видеоввода осуществляет запись полученного видеокадра в определенный участок памяти ЭВМ, перезаписывая содержимое участка через заданный интервал времени. Задержка времени на анализ кадров может приводить к потере некоторых кадров, снижая вероятность идентификации объектов.

Требуется определить зависимость вероятности $P(Id)$ идентификации объекта от количества кадров, выбранных случайным образом из множества N. Например, метка объекта распознана на k кадрах из N ($k>0$), тогда вероятность распознавания метки на одном кадре из N равна $\frac{k}{N}$.

Пусть анализируются m кадров из N, где $m \le N-k$. При $m > N-k$ вероятность идентификации равна 1, так как хотя бы один кадр, на котором метка распознана, попадает в состав выборки.

Обозначим событие R = «метка объекта распознана хотя бы на одном кадре из m». Оно является противоположным событию $\overline{R}$ = «метка объекта не распознана ни на одном кадре из m».

При $m=1$ вероятность события $\overline{R}$ равна:

$$P(\overline{R}) = \frac{N-k}{N}. \qquad (2.10)$$

При $m=2$:

$$P(\overline{R}) = \frac{N-k}{N} \cdot \frac{N-k-1}{N-1}, \quad (2.11)$$

так как вероятность $\overline{R}$ складывается из совместной вероятности событий «метка не распознана на первом выбранном кадре» и «метка не распознана на втором выбранном кадре». Поскольку выбор кадра осуществляется без возврата, то количество благоприятных исходов с каждым выбором уменьшается на единицу и общее количество исходов уменьшается на единицу. Отсюда следует:

$$P(R \mid m) = 1 - \prod_{i=1}^{m} \frac{N-k-i+1}{N-i+1}, \quad (2.12)$$

где $m \le N-k$, $N \ge k$.

В общем виде вероятность распознавания метки на m кадрах, случайным образом выбранных из N, при заданном k рассчитывается следующим образом:

$$P(R \mid m, N, k) = 1 - \prod_{i=0}^{m-1} \frac{N-k-i}{N-i}. \quad (2.13)$$

С увеличением m вероятность распознавания метки растет и при $i = N-k \Rightarrow m = N-k+1$ становится равной 1.

Пусть m – количество кадров, которое система распознавания способна анализировать из общего числа N, где $m \le N$. Данное количество определяется временем анализа отдельного кадра, то есть $m = t_O / T$. Округлим значение $m = \lfloor t_O / T \rfloor$, чтобы получить возможность использовать формулу 2.13.

При ограничении количества анализируемых кадров до m ($m < N$) вероятность $P(K_k)$ снижается пропорционально вероятности события $P(R \mid m, N, k)$:

$$P(K_k \mid m, N, k) = P(K_k) P(R \mid m, N, k). \quad (2.14)$$

Тогда вероятность идентификации объекта контроля рассчитывается:

$$P(Id \mid m, N) = \sum_{k=1}^{N} P(K_k) P(R \mid m, N, k). \quad (2.15)$$

В итоге связь между показателями T и $P(Id)$ определяется следующим образом:

$$P(Id) \approx \sum_{k=1}^{N} \left(P(K_k) \cdot \left(1 - \prod_{i=0}^{\lfloor t_O / T \rfloor - 1} \frac{N-k-i}{N-i} \right) \right). \quad (2.16)$$

Эта формула демонстрирует, что с уменьшением времени анализа кадра системой распознавания вероятность идентификации объектов контроля повышается.

Следует отметить, что меры, повышающие P_{right}, как правило, ведут к увеличению времени обработки кадра T. То есть, с одной стороны, увеличивается вероятность корректной обработки одного кадра, но с другой – снижается количество обработанных кадров. Количество кадров, которое следует обработать за единицу времени, то есть допустимое время анализа кадра, определяется условиями эксплуатации оптоэлектронной системы контроля.

Структура модели

Ниже приводится общая схема работы системы. На этой схеме показано, как именно разрабатываемая система объединяет алгоритм классификации и распознавания номеров в единую систему.

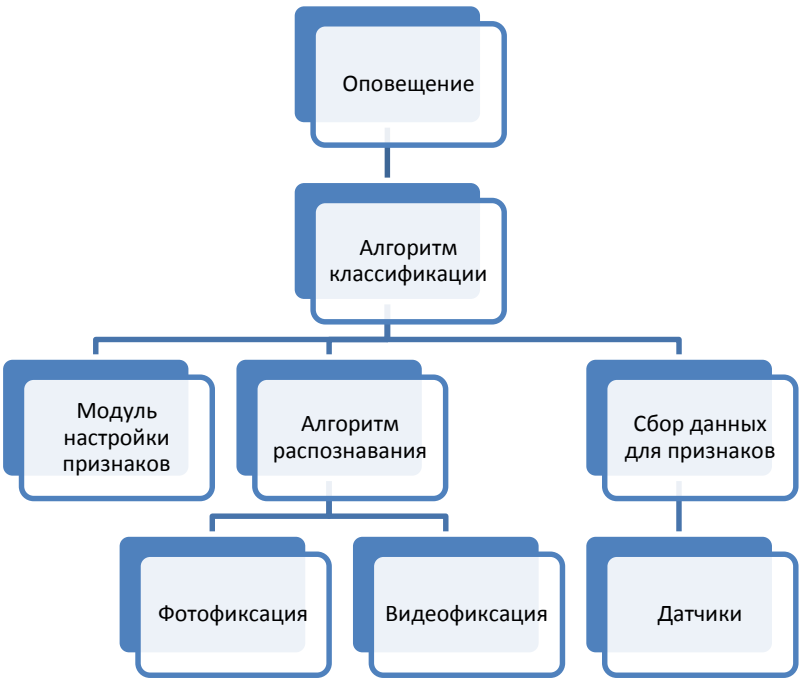


Рисунок 2.2 Структура разработанной модели.

1. На первом шаге производится сбор данных со всех сенсоров и датчиков. Производится фото и видеофиксация проезжающего автомобиля, а так же опрашиваются все датчики системы.

2. На втором шаге запускается алгоритм распознавания номеров, для идентификации автомобиля, а собранные с датчиков данные подготавливаются для подачи на вход алгоритму классификации.

3. На третьем шаге алгоритму классификации подается готовый вектор признаков, на основе которого система принимает решение об оповещении о нелегитимном событии. При этом используется программный модуль настройки признаков, в котором настроены все признаки, используемые в модели.

4. Далее производится переход к первому шагу.

Отметим, что данная модель удобно реализуется в рамках создания системы видеонаблюдения третьего поколения – она масштабируема, а потоки данных, передаваемые от уровня к уровню практически ничтожны, в

случае проведения первичной видеообработки на постах КПП, даже с учетом Real-Time реализации.

Выводы по второй главе

1. Данных из системы видеонаблюдения с распознаванием автомобильных достаточно для построения системы-классификатора, определяющую нелегитимную активность.
2. Критерии системы-классификатора возможно построить на основе шаблонов поведения людей.
3. По результатам исследования существующих подходов к определению нелегитимной активности был сделан вывод, о допустимости выявления нелегитимной активности на основе косвенных признаков.
4. Существующие алгоритмы распознавания разбиваются на стандартные блоки и их использование в системах реального времени допустимо.
5. С уменьшением времени анализа кадра системой распознавания, вероятность идентификации объектов контроля повышается.
6. Разработанную структуру модели допустимо реализовать в рамках рекомендаций по созданию систем видеонаблюдения третьего поколения.

Глава 3. Реализация системы выявления нелегитимных действий

Общие требования к реализации

Требования к реализации системы следуют из базовых требований к системе безопасности, работающей в режиме реального времени. Отметим, что система представляет собой комбинацию трех компонентов: камеры + датчики + пост (образуют «ячейку» разрабатываемой системы), а также координирующий сервер обработки данных. Также достаточно очевидным требованием является время работы без сбоев – для системы, обеспечивающей безопасность оно должно составлять не менее 99,7% процентов общего времени работы в год [54]. С учетом того, что система является программно-аппаратной, это означает, что аппаратная часть системы должна быть максимально отказоустойчивой. Отдельно следует отметить высокие требования, предъявляемые к качеству картинки для надежной работы алгоритма распознавания (о них было сказано в первой и второй главах). Далее по тексту требования будут изложены более подробно. Для начала перечислим общие требования, вытекающие из необходимости надежной работы программно-аппаратной системы в условиях машиностроения:

1. Система должна работать в режиме реального времени. Распознавание номеров, сбор данных, вычисление признаков, классификация – все должно происходить за время, в течение которого машина находится на КПП.
2. Система должна быть устойчива к перепадам температур.
3. Система должна требовать минимального обслуживания. В условиях российских предприятий очень трудно организовать регулярное и качественное обслуживание любых систем.

4. Система должна быть защищена от взлома и попыток внешнего воздействия. Как и любая система, обеспечивающая безопасность, разрабатываемая система должна иметь защиту от компрометации, поскольку на основе ее данных могут проводиться следственные мероприятия и приниматься серьезные кадровые решения (увольнения сотрудников).

5. Система должна максимально использовать существующую на предприятии инфраструктуру. Минимальный объем прокладки новых кабелей, использование существующих мощностей.

6. Система должна иметь резервирование данных. Вероятность пропажи данных, являющихся доказательной базой (а это – все данные системы) должна быть сведена к минимуму.

7. Система должна распознавать не менее 97% номеров автомобилей.

8. Время работы системы без сбоев в год – не менее 99.7%

Все требования пронумерованы, далее по тексту будут приводиться ссылки как на «Требование 1» и т.п.

Вышеперечисленные требования являются достаточно стандартными для промышленных систем. Выполнение части из них достигается стандартными подходами – использованием морозоустойчивой аппаратуры, защищенных протоколов передачи данных и т.д. Небольшого объема сервисного обслуживания такой системы можно добиться, используя самоочищающиеся кожухи для камер, а высокой производительности системы распознавания автомобильных номеров – используя многопоточную обработку кадров.

Архитектура предлагаемой системы и формулировка технических требований

На основании общих требований была разработана распределенная архитектура системы, которая обеспечивает высокую защищенность системы и низкую нагрузку на каналы передачи данных.

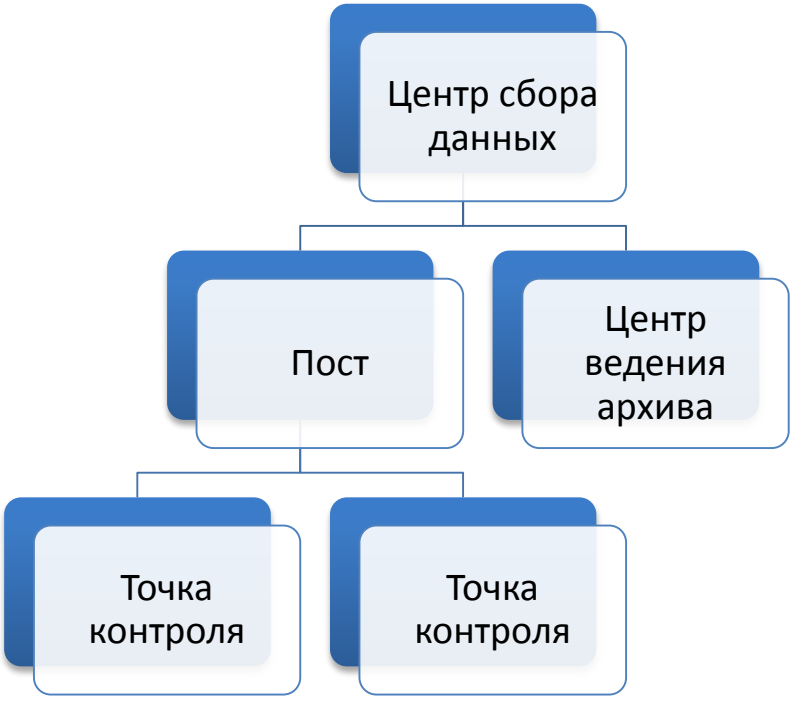


Рисунок 3.1 Архитектура системы

Особенности архитектуры:

- Высокая защищенность, за счет реализации кеширования данных в точках контроля. В том случае, если центр сбора данных выходит из строя, данные записываются локально до тех пор, пока не появится связь с выделенным сервером
- За счет реализации веб-интерфейса центра сбора данных в рамках стандарта HTML 5.0 интерфейс системы доступен с любого современного устройства, что позволяет отслеживать события с различных мобильных устройств

- Центр ведения архива вынесен в отдельную сущность с целью обезопасить данные от возможной кражи и компрометации. В нем применяется трехкратное дублирование данных для повышения надежности.
- Обмен данными производится при помощи защищенного протокола SSL

Отметим, что данная архитектура способствует выполнению требований 1 (за счет распределенности) и требования 4 (за счет единого центра обработки, обеспечивающего защищенность данных и защищенного с помощью протокола SSL обмена данными). Также поскольку обмен данными ведется по стандарту Ethernet (витая пара) по существующим на предприятии каналам связи, обеспечивается выполнение требования 5.

Опишем далее все составные части архитектуры системы.

Точки контроля – набор телекамер, установленных на одном из КПП:

1.1. одна или несколько телекамер распознавания номеров автомобилей;

1.2. одна или несколько телекамер наблюдения (общая панорама места, место водителя, содержимое кузова и т.д.);

1.3. различные датчики – датчики открытия-закрытия шлагбаумов, магнитные датчики проезда автомобилей.

2. **Посты** – пункты сбора данных с одной или нескольких точек контроля, объединяющие:

2.1. компьютер на платформе Wintel и дополнительные аппаратные средства обеспечения непрерывной работы;

2.2. программное обеспечение распознавания номеров автомобилей и предварительной обработки видео.

3. **Центр сбора данных** – место объединения и хранения данных о движении транспорта с постов контроля, содержащее:

3.1. высокопроизводительный компьютер-сервер и доп. аппаратные средства обеспечения непрерывной работы и защищенного хранения данных;

3.2. программное обеспечение СУБД «Microsoft SQL Server» для хранения данных;

3.3. программное обеспечение для управления данными;

3.4. программное обеспечения АРМ пользователя на основе веб-интерфейса.

3.5. классификатор нелегитимных событий

3.6. систему оповещения на основе рассылки e-mailсообщений

4. **Центр ведения архива** – место объединения и хранения видеоданных с постов контроля, сервер содержит:

4.1. высокопроизводительный компьютер-сервер и доп. аппаратные средства обеспечения непрерывной работы и защищенного хранения данных (может быть использован сервер центра сбора данных);

4.2. программное обеспечение СУБД Microsoft SQL Server для хранения данных (может быть использована СУБД центра сбора данных);

4.3. программное обеспечение для управления данными и организации доступа;

4.4. массив легитимных и нелегитимных событий;

4.5. программное обеспечения АРМ пользователя на основе веб-интерфейса.

Связь между точками контроля и постами осуществляется посредством проложенных коаксиальных, оптоволоконных, витой пары и силовых кабелей, а между постами и центрами системы –посредством имеющейся на

предприятии сетевой инфраструктуры. В тех случаях, когда подключение поста к имеющейся сети невозможно, связь организуется посредством технологий VPN через GSM UMTS и Wi-Fi.

Пользователи работают с системой посредством рабочих компьютеров, также подключенных к сети предприятия посредством веб-браузера. Это избавляет от необходимости использования стороннего программного обеспечения, а также обеспечивает выполнения требования 5.

Распределенность данной системы – один из ее основных плюсов, позволяющий снизить нагрузку на центральный сервер, а также повысить надежность и масштабируемость системы. Также это позволяет гибко добавлять новые модули и датчики в уже существующую систему. Единицей системы является пост – к нему подключены камеры системы, а также различные датчики. Серверная часть не имеет ограничений на количество одновременно подключенных постов.

Одной из основных потенциальных проблем является поломка и порча оборудования, поэтому все камеры, ЭВМ и другие части системы ставятся в максимально защищенных и неудобных для воров местах. Все кабели прокладываются под землей в защищенных трубах, чтобы избежать кражи кабелей.

Ниже приведем технические требования, предъявляемые к камерам системы, связанные с ограничениями и требованиями программного модуля распознавания номеров.

Требования к телекамерам распознавания

- Общая характеристика – аналоговая черно-белая высокочувствительная камера с модулем ИК-подсветки в защищенном корпусе.
- Стандарт видеосигнала: CCIR (PAL).

- Разрешение не ниже 600 линий (при меньшем разрешении не будет работать распознавание [15,17,18])
- Чувствительность не хуже 0,0005 лк (для распознавания в ночное время)
- Возможность установки фиксированного времени экспозиции 1/1000 сек (для уменьшения шума на изображении)
- Подсветка зоны контроля – инфракрасная или видимым светом (для обеспечения распознавания в ночное время)

Требования к телекамерам наблюдения (вспомогательные камеры):

- Общая характеристика – аналоговая цветная высокочувствительная камера «день-ночь» с модулем ИК-подсветки в защищенном корпусе
- Стандарт видеосигнала: CCIR (PAL) (для стандартного подключения к ЭВМ поста, обеспечивает требование 5)
- Поддержка режима «день-ночь», автоматический ИК-фильтр (для получения изображения в ночное время)
- Подсветка зоны контроля – инфракрасная или видимым светом

Вышеперечисленные требования – стандартны для систем распознавания. В противном случае не будет достигаться достаточная точность распознавания [2]. Программный модуль, описанный во второй части может работать с любыми изображениями, однако при использовании оборудования не обеспечивающего вышеперечисленных требований, не будет соблюдаться требования 1, 2 и 4. Например, при меньшем количестве линий процент распознанных номеров снижается на треть и уже не удовлетворяет требованию 7. Отметим ключевую особенность – использование инфракрасной подсветки для улучшения распознавания ночью. Так же в системе используются камеры с качественным АЦП внутри, незначительно искажающим изображение. После тщательного анализа

рынка доступных камер оказалось, что этим требованиям отвечают только камеры компании Digivi 9000 серии [1,2].

Отметим, что все камеры размещаются в специальных защитных кожухах для выполнения требования 4 и требования 8. Также кожухи объективов оборудованы дворниками и подогревом стекла, для обеспечения чистоты изображения вне зависимости от погодных условий (для тех же требований). Расположение камер и параметры объективов выбираются исходя из следующих принципов:

- Максимально четкое изображение для распознавания номеров. Поскольку камеры обладают лимитированным фокусным расстоянием, крайне важно расположить камеру так, чтобы в точке останова машины на КПП картинка с номером была максимально четкой, иначе не выполняется требование 7.
- Наиболее вандалозащищенное положение. Поскольку на большинстве предприятий процветает вандализм и порча оборудования, то камеры располагаются в максимально недоступных для вандалов местах (требование 4).

Далее приведем технические требования, предъявляемые к компьютеру на посту. Одним из основных требований предъявляемым к современным системам видеонаблюдения является стандартность компонент – иначе дальнейшее обслуживание системы может быть затруднено, как и поиск запасных частей. Обозначим это требование как требование 9. Технические требования на производительность ПК были получены исходя из оценки требований модуля распознавания.

- Общая характеристика – стационарный компьютер стандарта x86 в корпусе с доп. оборудованием для подключения не более 16 телекамер (требования 5 и 9).
- Процессор не слабее IntelCore 2 Duo 2ГГц или Intel i3

- Объем ОЗУ не менее 2 Гб.
- Наличие устройства видеозахвата на основе Conexant Bt878 или Phillips SAA7130, с числом каналов не менее числа камер поста.
- Системное ПО: Windows 7 Starter (требование 9).

Этот компьютер подключен к камерам наблюдения наиболее близко расположенным к данному посту. Пост получает данные с камер, обрабатывает согласно алгоритмам, создает локальную резервную копию данных и передает данные далее, в Центр сбора данных.

Опишем программное обеспечение, установленное на посту:

- Обрабатывает видеопоток данных
- Фиксирует факты проезда автомобилей
- Распознает номера, проехавших автомобилей
- Ведет локальный видео и фото архив.
- Передает данные о фактах проезда и с датчиков в единый центр данных.

Разработанное программное обеспечение представляет собой программный модуль, написанный на платформе Microsoft .Net. Данное решение было выбрано, поскольку платформа .Net значительно ускоряет срок разработки и многие системные функции (такие как доступ к видеоданным), на данной платформе проще реализовать. Данный модуль скомпилирован в виде исполняемого файла для платформы wintel, который работает 24/7 и обеспечивает выполнение вышеперечисленных функций. Ниже приводится схема ключевых блоков данного модуля:

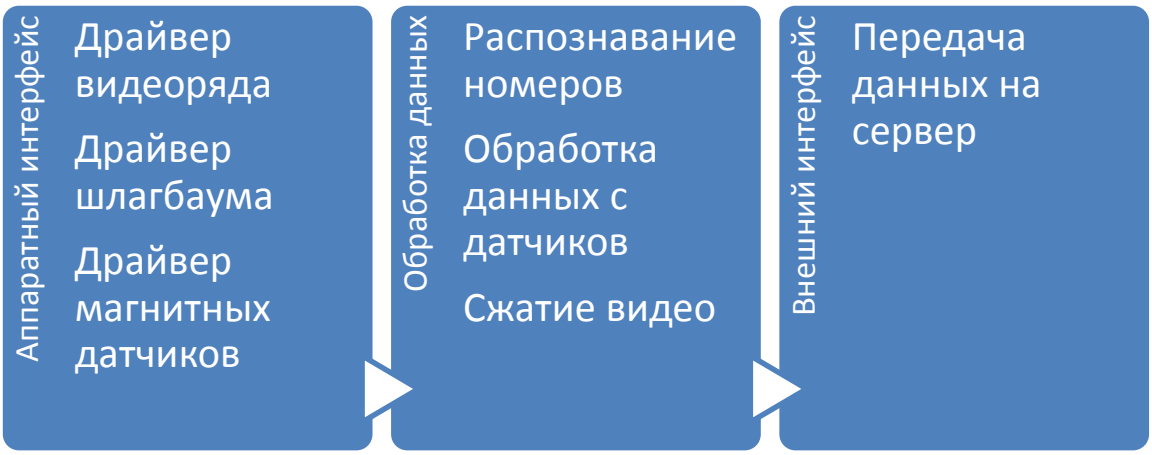


Рисунок 3.2 Схема работы программного модуля на посту.

Опишем блоки программного обеспечения:

1. Аппаратный интерфейс состоит из классов, реализующих управление шлагбаумом, магнитными датчиками и потоком видеоряда. Для взаимодействия с драйверами был написан класс DriverInterface, реализующие работу со всей аппаратной частью. Он состоит из 12348 строк.
2. Модуль обработки данных, состоит из трех подмодулей. Первый реализует алгоритм распознавания номеров и подробно описан в [1,2]. Второй обрабатывает данные с датчиков и подготавливает их к отправке на сервер. Третий сжимает видео по стандарту Motion Picture Experts Group 4 в целях экономии места на диске и уменьшения нагрузки на каналы передачи данных.
3. Третий модуль занимается отправкой данных на сервер. Более подробно он описан ниже.

Для обеспечения стабильной работы системы крайне важно обеспечить надежность механизма передачи данных в центр сбор данных и реализовать локальное кеширование данных. Для реализации этого механизма был написан класс RemoteServerExporter, его исходный код

приводится ниже. Для облегчения читабельности кода средствами Microsoft Visual Studio код был переведен с платформы .Net на VB.Net

```
1.  Public Class RemoteServerExporter
2.    Implements IEventsSaver
3.    Private _source As IEventsSource
4.    Private _enabled As Boolean
5.    ….
6.    Private _toSend As New List(Of EventDataPack)
7.    Sub New(storage As SettingsStorage, logger As Logger)
8.      If storage Is Nothing Then Throw New ArgumentException
9.      If logger Is Nothing Then Throw New ArgumentException
10. //подготовка переменных и объектов
11.       _logger = logger
12.       _storage = storage
13.       _use = _storage.CreateBooleanSetting("Use", False)
14.       _placeInfo = _storage.CreateStringSetting("PlaceInfo", "Post1")
15.       _placeGuid = _storage.CreateStringSetting("PlaceGuid", "post-1")
16.       _sendQueue = _storage.CreateIntegerSetting("SendQueue", "50")
17.       _interface = New KerberUploaderClient(_storage)
18.       _thread = New Thread(AddressOf SendToWeb)
19.       _thread.IsBackground = True
20.       _thread.Name = "SendToWeb"
21.       _thread.Start()
22.   End Sub
23.   Public Sub ConnectToSource(source As IEventsSource) Implements IEventsSaver.ConnectToSource
24.     If _source IsNot Nothing Then
25.       RemoveHandler _source.NewEvent, AddressOf NewEventHandler
26.     End If
27.     _source = source
28.     If _source IsNot Nothing Then
29.       AddHandler _source.NewEvent, AddressOf NewEventHandler
30.     End If
31.  End Sub
32.   Public Property Enabled As Boolean Implements IEventsSaver.Enabled
33.     Get
34.       Return _enabled
35.     End Get
36.     Set(value As Boolean)
37.       _enabled = value
38.     End Set
39.   End Property
40.   Public ReadOnly Property Name As Object Implements IEventsSaver.Name
41.     Get
42.       Return "Exporter"
```

```
43.         End Get
44.       End Property
45.       Private Sub NewEventHandler(from As IEventsSource, data As EventDataPack)
46.     //для решения проблем при совместной синхронизации используются блокирующие семафоры
47.         SyncLock Me
48.           If _use.Value = False Then Return
49.           SyncLock _toSend
50.             _toSend.Add(data)
51.             If _toSend.Count > _sendQueue.Value Then
52.               _toSend.RemoveAt(0)
53.             End If
54.           End SyncLock
55.         End SyncLock
56.       End Sub
57.     //процедура отправки данных на веб-сервер
58.       Private Sub SendToWeb()
59.         Do
60.           Thread.Sleep(10)
61.           Dim data As EventDataPack = Nothing
62.           Dim count As Integer
63.           SyncLock _toSend
64.     //проверка очереди
65.             If _toSend.Count > 0 Then data = _toSend(0)
66.             count = _toSend.Count
67.           End SyncLock
68.           If data IsNot Nothing Then
69.             Try
70.     //подготовка объекта данных – приводится сокращенно
71.               Dim ke = New KerberEvent
72.               ke.ID = Guid.NewGuid.ToString
73.               ke.PlaceGuid = _placeGuid.Value
74.               Dim ms1 As New System.IO.MemoryStream
75.               data.BestFrame.Work.Save(ms1, Imaging.ImageFormat.Jpeg)
76.     //заполнение объекта данными – приводится сокращенно
77.                 _interface.UploadFrame(kf2, kf2.ID + ".jpg", ms2.ToArray)
78.               End If
79.               _interface.UploadFrame(kf1, kf1.ID + ".jpg", ms1.ToArray)
80.               _interface.UploadEvent(ke)
81.     //смена семафора
82.               SyncLock _toSend
83.                 _toSend.RemoveAt(0)
84.               End SyncLock
85.     //отладочные данные
86.               _logger.AddInformation("Отправлено: " + data.Plate + ", осталось: " + count.ToString)
87.             Catch ex As Exception
```

```
88.                     _logger.AddWarning("Ошибка отправки, ожидание: " + ex.Message + ", осталось: " +
count.ToString)
89.                     Thread.Sleep(5000)
90.                 End Try
91.             End If
92.         Loop
93.     End Sub
94.     Sub New()
95.     End Sub
96. End Class
```

Как видно из реализации класса, для реализации кеширования и пересылки данных используются мониторы Хоара (конструкция SyncLock) реализующие асинхронный доступ к серверу. Поскольку архитектура подразумевает одновременное использование сервера несколькими клиентами используется механизм синхронизации, для исключения возможности блокировки сервера несколькими процессами.

Также исходя из общих требований были сформулированы требования, предъявляемые к компьютеру на посту:

- Бесперебойная работа, даже в случае потери питания. Для этого компьютер подключается к источнику бесперебойного питания.
- Защищенность от взлома. Данные архива – конфиденциальные, получение доступа к ним злоумышленника – недопустимо. Для этого используются стандартные средства шифрования Windows, уровня Bitlocker.

Ниже приведены технические требования, предъявляемые к компьютеру в центре ведения архива (требования уточняются для новых пунктов).

- Общая характеристика – стационарный компьютер стандарта x86.
- Процессор не хуже Intel i7 3ГГц, не менее 4 физических процессорных ядер.

- Объем ОЗУ не менее 8 Гб.
- Дисковая подсистема RAID 1 или 0+1 объемом не менее 2 Тб (для обеспечения требования 6)
- СУБД: SQL Server 2008\2012 Express (для ведения базы видеоархива).

Центр ведения архива/сервер:

- Надежно сохраняет данные о фактах проезда в виде видеоряда с привязкой к распознанным фактам проезда, т.е. возможно найти все моменты проезда машины с заданным государственным номером и посмотреть события, которые были зафиксированы камерой видеонаблюдения
- Хранит данные о проездах за период от трех месяцев.
- Защищен от взлома с помощью алгоритмов шифрования.
- Реализует классификацию и оповещение

Разработанное программное обеспечение представляет собой программный модуль, написанный на платформе Microsoft .Net. Данное решение было выбрано, поскольку платформа .Net значительно ускоряет срок разработки и многие системные функции (такие как доступ к видеоданным), на данной платформе проще реализовать. Данный модуль скомпилирован в виде исполняемого файла для платформы wintel, который работает 24/7 и обеспечивает выполнение вышеперечисленных функций. Ниже приводится схема ключевых блоков данного модуля:

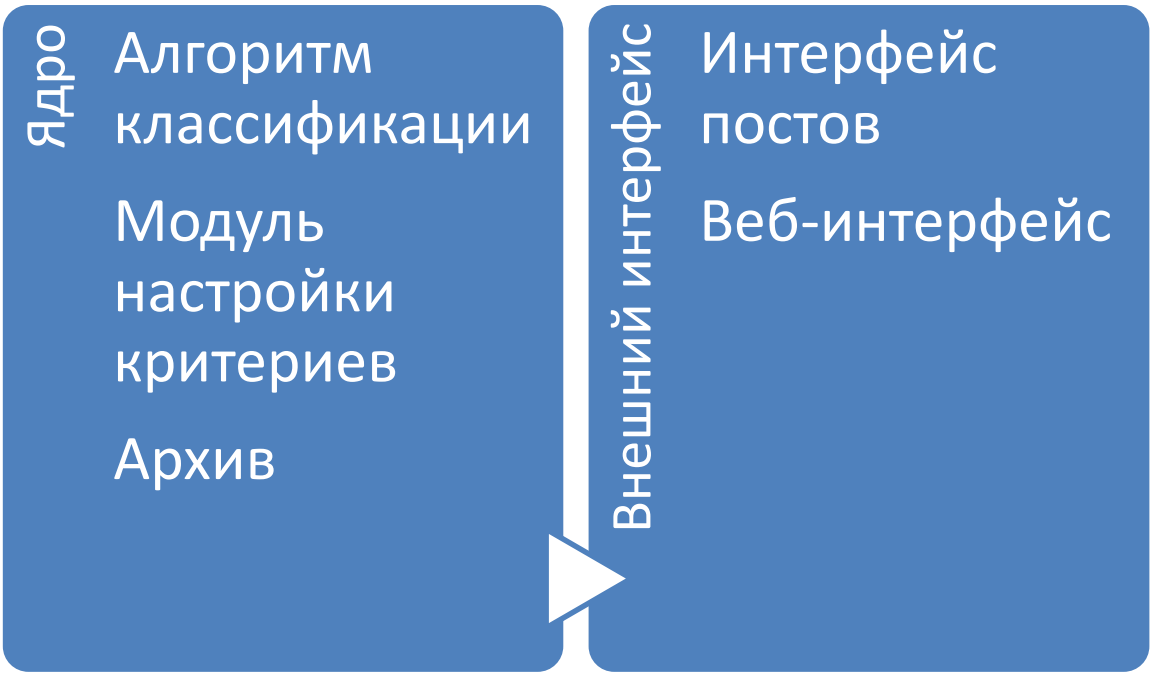


Рисунок 3.3 Схема работы сервера.

Опишем блоки программного обеспечения:

1. Ядро состоит из трех компонентов. Первый реализует описанный во второй главе алгоритм классификации. Второй позволяет на си-подобном псевдокоде настраивать список критериев нелегитимности (код аналогичен использованному во второй главе при описании критериев). Также существует архив написанный на MS SQL (для соблюдения требования 5). Ядро состоит из 32741 строки.
2. Внешний интерфейс, состоит из двух подмодулей. Первый реализует обмен данными между постами и сервером [54]. Второй является веб-сервером на ASP.Net, реализующим веб-интерфейс системы. Скриншоты веб-интерфейса приводятся ниже.

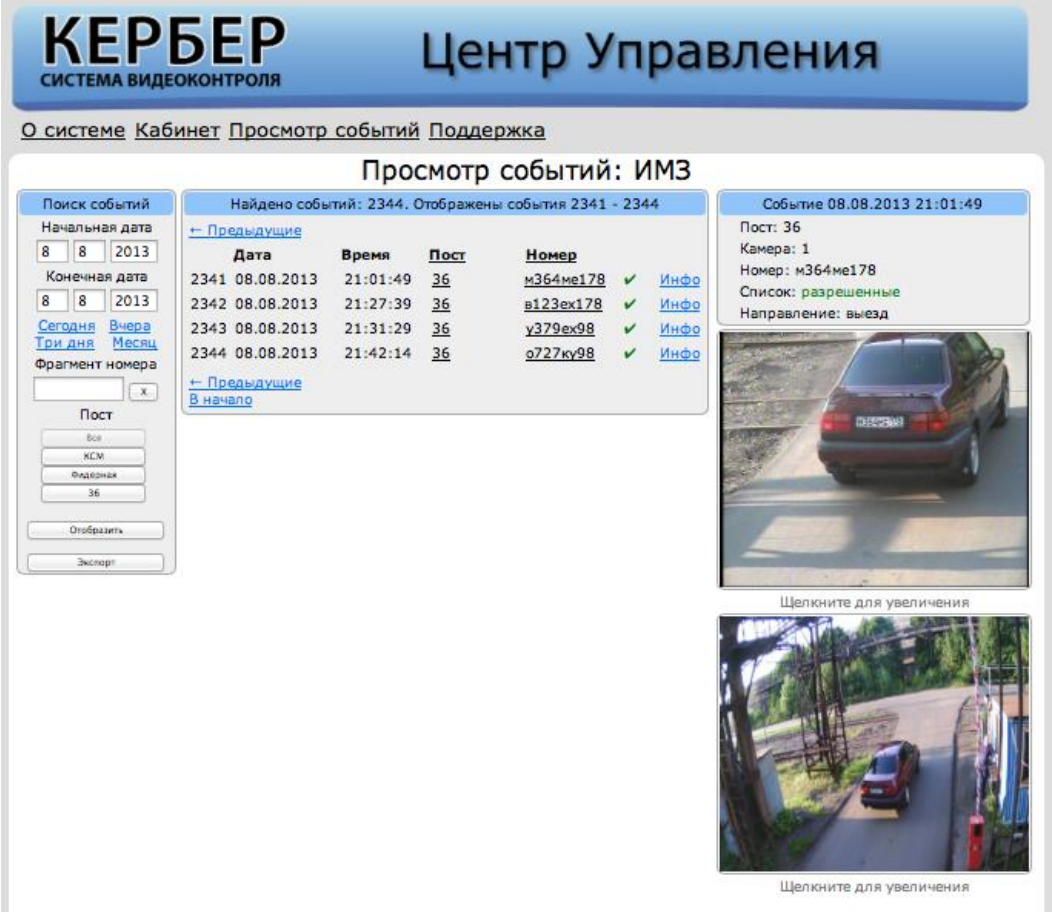


Рисунок 3.4 Веб-интерфейс реализованной системы

Рисунок 3.5 Главное окно системы

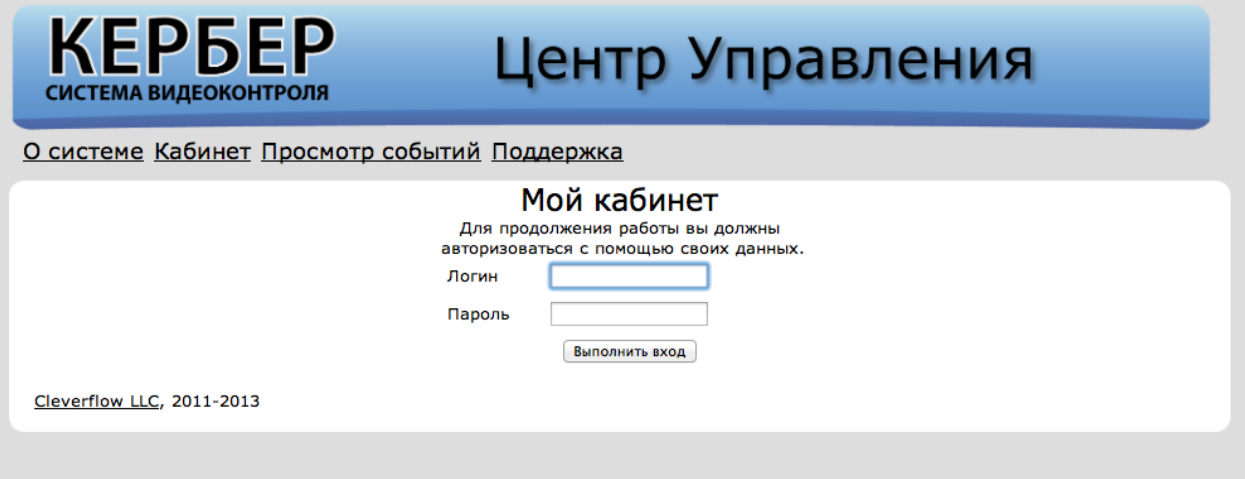


Рисунок 3.6 Окно входа в систему

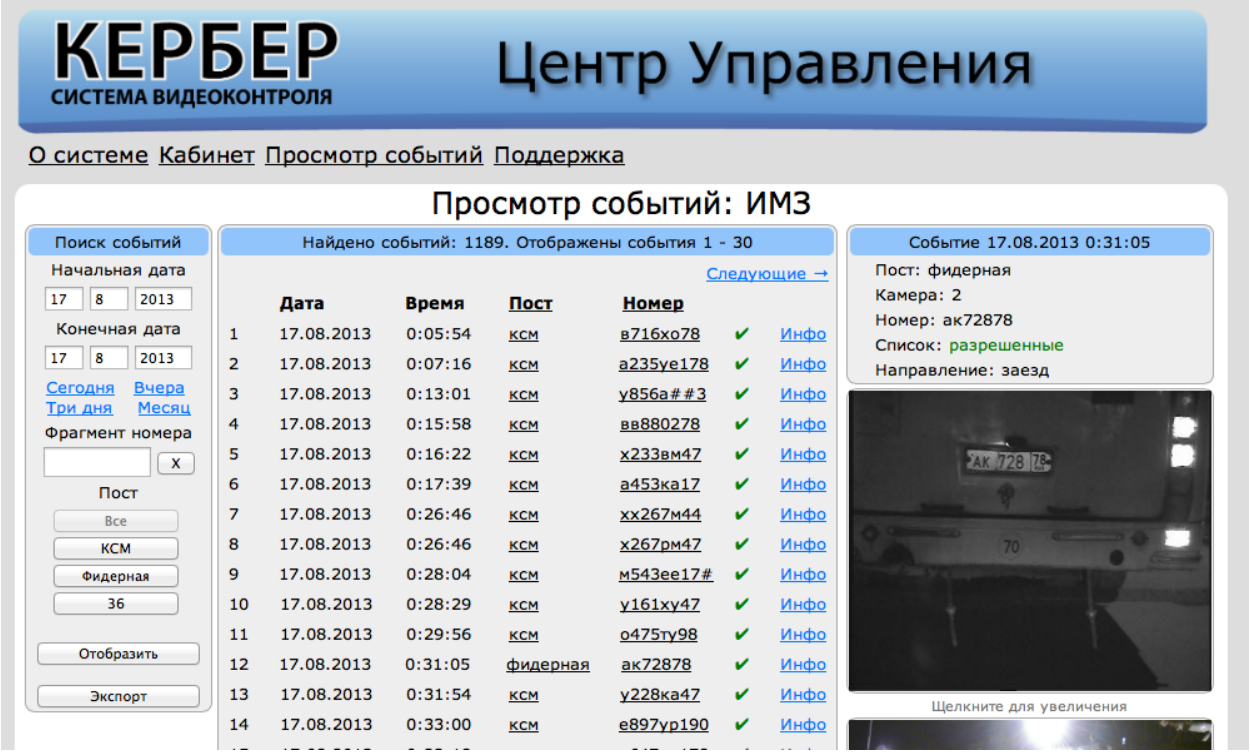


Рисунок 3.7 Система в работе

Исходя из описанных выше условий к системе коммуникаций предъявляются следующие требования:

- Требования к пропускной способности канала связи пост-центр: 512 кбит\с, если видеоданные с поста не архивируются, 5 Мбит\с, если архивируются.
- Требований к задержкам каналов связи не предъявляется.

При этом используются стандартные каналы – витая пара, оптоволокно, коаксиальный кабель. Возможно также использование WiFi, при отсутствии возможности физического подключения, поскольку все используемые протоколы для обмена данными являются стандартными.

Итоговое описание реализации

Реализация системы выполнялась на платформе Wintel с использованием фреймворка Microsoft .Net 4.0. Обмен данными между модулями системы реализован по защищенным криптографией протоколам

SSl и исходя из опасности внезапного прерывания связи между объектами системы. Поэтому в системе повсеместно используется кеширование и дублирование всей информации.

Оповещение о нелегитимных событиях производится путем рассылки от встроенного e-mail сервера писем лицам, отвечающим за безопасность, а также sms рассылки через sms-шлюз операторов.

Все объекты системы смонтированы вандалозащищенным образом, а также по возможности сокрыты от посторонних глаз. Например, все компьютеры на постах установлены в вандалозащищенные ящики.

Потоки данных

Отдельно опишем, какими информационными потоками обмениваются различные части системы. На схеме ниже приведена схема потоков данных внутри разработанной системы.

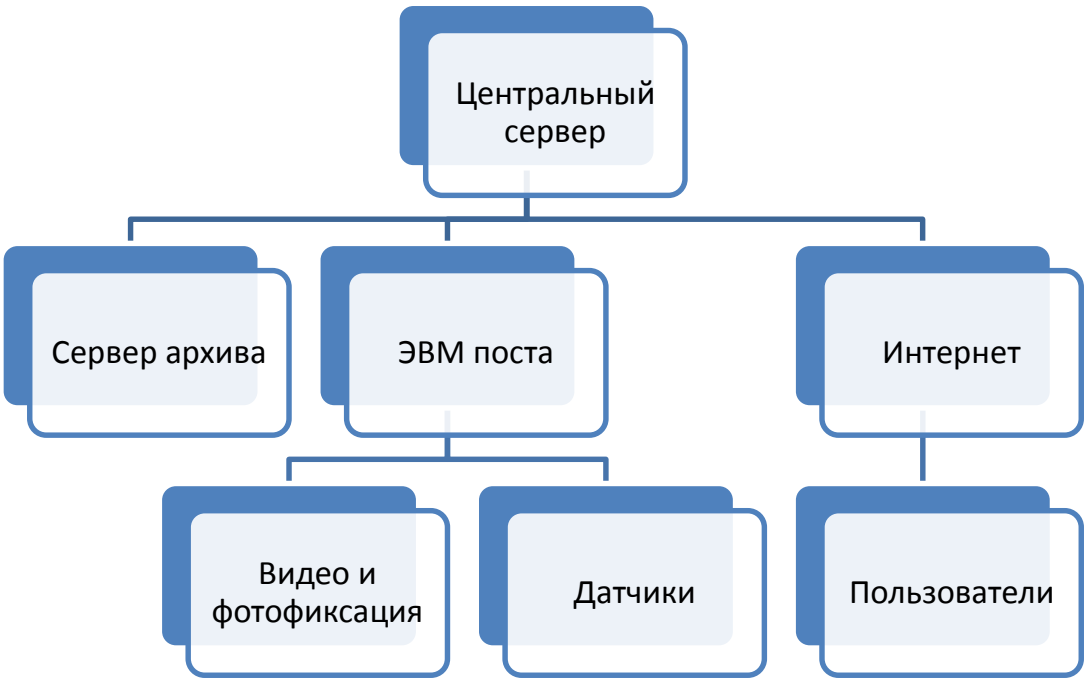


Рисунок 3.8 Потоки данных внутри системы

Ниже приводится описание потоков данных внутри системы:

- Датчики передают данные в ЭВМ поста. Объем передаваемых данных по шине USB 2.0 Full-speed: 50 Кбайт/сек, 0.1% пропускной способности каналов
- Видео и фото передается с камер в ЭВМ поста. Объем передаваемых данных по шине USB: 100 Мбит/сек, 30-40% пропускной способности канала
- Сервер архива двусторонне обменивается данными с Центральным сервером для хранения данных. Объем передаваемых данных по Gigabit Ethernet: 500 Мбит/сек, 50% пропускной способности канала

- ЭВМ поста обменивается данными с Центральным сервером для обработки данных. Объем передаваемых данных по Gigabit Ethernet (видео и фото предварительно сжимаются): 50 Мбит/сек, 5% пропускной способности канала
- Центральный сервер выводит свои данные в сеть Интернет и локальную сеть предприятия. Объем передаваемых данных по Gigabit Ethernet (в расчете на 2000 пользователей): 1 Мбит/сек, 1% пропускной способности канала

Основная решенная задача – предварительная обработка и сжатие фото и видео на ЭВМ поста, позволившая снизить нагрузку на каналы связи на 80%. При выключении предварительного сжатия данных сеть предприятия аварийно перезагружается.

Результаты эксплуатации

Разработанная система была внедрена в ОАО «Ижорские заводы» в следующих условиях:

- ОАО «Ижорские заводы» является одним из крупнейших заводов в России, занимающимся тяжелым машиностроением
- 94% процента потока, проезжающего через КПП заводы – большегрузный транспорт
- Объем краж в год составляет около 105 миллионов рублей
- Система была установлена на трех КПП (на всех, через которые проходит большегрузный транспорт)
- Количество краж в месяц с использованием крупных автомобилей – не менее 30, около 25 из них выявляется постфактум
- Срок эксплуатации – 5 месяцев
- Количество выездов машин в месяц – не менее 300 000

Результаты эксплуатации приведены в следующей таблице 3.1. Система была установлена в октябре 2012 года. С октября 2012 года в течение пяти месяцев собиралась статистика, а каждые 24 часа оператор корректировал массив легитимных выездов в системе.

За это время система позволила выявить 100% случаев нелегальных вывозов, из тех, пропажи которых были обнаружены.

Таблица 3.1 Результаты эксплуатации системы

Параметр	Значение	Описание
Количество выехавших машин	1 754 874	Количество машин прошедших хотя бы через одно КПП на территории завода с начала эксплуатации системы
Количество выявленных попыток нелегитимных действий	145	Всего выявлено – 145, из них 7 – попытка вывоза продукции, 138 – несоблюдение регламента и/или установленного порядка
Количество невыявленных попыток нелегитимных	0	Успешных краж с использованием автотранспорта на территории промышленной

действий		площадки с момента установки системы не зафиксировано

Результаты использования системы достаточно многообещающие, однако следует отметить:

- Помимо установки системы на предприятии была заменена половина охранников, обслуживающих КПП
- Также не следует забывать про эффект запугивания, который оказывает информация об установке подобной системы на сотрудников предприятия

Но даже с учетом этих оговорок, результаты работы системы можно оценить, как отличные. На текущий момент руководство предприятия решило использовать нашу систему на всех принадлежащих ней заводов.

Выводы по третьей главе

1. Реализованная в рамках диссертационной работы система «Цербер» выявляет попытки краж в условиях предприятия с процентом 97%
2. Таким образом, для выявления нелегитимной активности достаточно использовать косвенные признаки, например долгое время осмотра автомобиля и т.д., без использования требующих больших затрат алгоритмов машинного зрения
3. Предложенная архитектура системы показала высокую надежность работы
4. Особую ценность представляет собой устойчивость системы независимость различных частей системы друг от друга – в случае выхода из строя одной точки наблюдения и/или центрального сервера, работы системы не будет нарушена
5. Использование выбранных в процессе выполнения диссертационной работы признаков показало нелегитимной активности показало высокую эффективность при работе в реальных условиях

Заключение

Таким образом, реализованная в рамках данной работы система, основанная на комбинации системы видеофиксации и классификации нелегитимных действий, позволила значительно повысить процент выявляемых на предприятии краж. Также объем потерь из-за краж снизился на порядок – 15 миллионов рублей, вместо 150-190 миллионов. Направлениями дальнейшей работы можно назвать следующие:

- Изучение возможности использования других признаков. Выбрано восемь достаточно сложных для вычисления признаков (для вычисления половины из них требуется проведения достаточно сложных статистических подсчетов для вычисления среднего времени).
- Дополнительного изучения требует вопрос о возможности получения сравнимой эффективности с другим набором признаков (другая программная настройка).
- Изучение возможности автоматического выделения признаков в системе. Это позволило бы значительно снизить сложность интеграции и первичной настройки системы.
- Дальнейшее развитие модуля распознавания номеров, в частности изучение вопроса его первичной калибровки в автоматическом режиме.
- Изучение возможности контроля не только за автотранспортом, но и за людьми, в целях предотвращения мелких краж.
- Более плотная интеграция системы с электронным документооборотом на предприятии с целью улучшения электронного документооборота, в частности отдела пропусков.

Литература

1. Абрамов Н.А. Компьютеризированная система контроля трафика на крупных предприятиях (система «Цербер») [Текст] Труды ИСА РАН. Том 62, выпуск 3. – с. 3-10.

2. Абрамов, Н.А., Кошелев И.И., Применение стереоскопического эффекта для расчета динамических характеристик движущихся автотранспортных средств и схема распознавания номеров в системе «Цербер» [Текст]: Международный технико-экономический журнал, №2, 2013, с. 79-84.

3. Абрамов Н.А., «Две задачи оптимального управления технологическим процессом» [Текст] Труды Института системного анализа РАН, Том 53 (1), 2010, стр. 124-131.

4. Абрамов Н.А., «Об одном критерии в задаче выбора оптимального маршрута» [Текст] Труды Института системного анализа РАН 2010. Т. 32 (2), 2010, стр. 316-321.

5. Абрамов Н.А., Качалин А.И. «Выбор моделей распространения ВПО при разработке модели глобальной сети», Методы и средства обработки информации. [Текст] Труды третьей Всероссийской научной конференции/Под ред. Л.Н. Королева. – М.: Издательский отдел Факультета ВМиК МГУ им. М.В. Ломоносова, 2009., стр. 433-438.

6. Абрамов Н.А., «Построение модели влияния трафика вредоносного программного обеспечения на трафик в глобальной сети» [Текст] Сборник тезисов лучших дипломных работ 2010 года/Сост.: Позднеев А.В., Шевцова И.Г. – М.: Издательский отдел Факультета ВМиК МГУ им. М.В. Ломоносова; МАКС Пресс, 2010., стр. 104-106.

7. Абрамов, Е. С. Моделирование систем распознавания изображений на примере печатных текстов [Текст] : дис. … канд. техн. наук : 05.13.01 / Е. С. Абрамов. – Санкт-Петербург, 2006. – 140 с.

8. Арлазаров, В. Л. Адаптивное распознавание символов. [Текст] / В. Л. Арлазаров, А. Д. Астахов, В. В. Троянкер, [и др.] // Интеллектуальные технологии ввода и обработки информации. Сборник трудов. – М.: Институт системного анализа РАН, 1998. – С. 39–56.

9. Арлазаров, В. Л. Алгоритмы распознавания и технологии ввода текстов в ЭВМ. [Текст] / В. Л. Арлазаров, О. А. Славин // Информационные технологии и вычислительные системы. – 1996. – № 1. – С. 48–54.

10. Арлазаров, В. Л. Распознавание строк печатных текстов [Электронный ресурс] / В. Л. Арлазаров, П. А. Куратов, О. А. Славин; Cognitive Technologies. – Электрон. дан. – М., 2000. – Режим доступа : http://www.cognitive.ru/innovation/sbornic2/. – Загл. с экрана.

11. Бондаренко А. В. , Исследование подходов к построению систем автоматического считывания символьной информации [Электронный ресурс] / В. А. Галактионов, В. И. Горемычкин [и др.]; ИПМ им. М. В. Келдыша РАН. – Электрон. дан. – М., 2003. – Режим доступа : http://www.keldysh.ru/papers/2003/prep44/prep2003_44.html. – Загл. с экрана.

12. Воскресенский, Е.М. Параметрическая и структурная адаптация систем распознавания текстовых меток на видеоизображениях: дис. … канд. тех. наук: 05.13.01 / Воскресенский Евгений Михайлович; [Ин-т менеджмента и информ. технологий]. - Рыбинск: 2010. - 16 с.: ил.; 21 см.

13. Горский, Н. Д. Распознавание рукописного текста: от теории к практике [Текст] / Н. Д. Горский, В. А. Анисимов, Л. А. Горская. – СПб. : Политехника, 1997. – 126 с.

14. Журавлёв, Ю. И. Распознавание. Математические методы. Программная система. Практические применения [Текст] : монография / Ю. И. Журавлёв, В. В. Рязанов, О. В. Сенько. – М.: ФАЗИС, 2006. – 176 с.

15. Иванов Д. Знак качества [Электронный ресурс] // Broadcasting. Телевидение и радиовещание. – 2006, #1. – URL: http://www.broadcasting.ru/articles2/Oborandteh/sign_qualit (дата обращения 30.08.2012).

16. Икрамов К.С. Методы и устройства оценки изменений информационных параметров сигналов изображений в системах телевизионного контроля объектов : диссертация ... кандидата технических наук : 05.12.04 / Икрамов Кобул Сабирович; [Место защиты: Моск. техн. ун-т связи и информатики] – Москва, 2011. – 205 с. ил.

17. Комар В.Г. Количественные критерии качества изображения для оценки кинематографических систем [Электронный ресурс] // ТКТ, 2000, №10. – URL: http://barsic-cat.chat.ru/tkt/10_2000/komar.htm (дата обращения 16.03.2012).

18. Кривошеев М. И., Мкртумов А. С., Федунин В. Г. Качество изображения и измерения в цифровом телевидении [электронный ресурс]// 625. 1999. № 1. – С.72-75. – URL: http://rus.625-net.ru/625/1999/01/krivosheev.htm (дата обращения 16.03.2012).

19. Макаревич, О. Б. Метод оптического распознавания печатного текста на основе псевдодвумерных скрытых марковских моделей [Текст] / О. Б. Макаревич, Л. К. Бабенко, М. В. Аникеев [и др.] // Научно-практическая конференция "Информационная

безопасность", сборник трудов. – Таганрог: Изд-во ТРТУ, 2002. – С. 195–197.

20. Монич Ю.И., В.В. Старовойтов. МЕРА ОЦЕНКИ РЕЗКОСТИ ЦИФРОВОГО ИЗОБРАЖЕНИЯ [электронный ресурс] // ДОКЛАДЫ БГУИР. – 2011. – С. 80–85. – URL: http://www.bsuir.by/m/12_104571_1_69801.pdf (дата обращения 01.06.2012).

21. Руцков М.В.; Система считывания автомобильных номеров [Электронный ресурс] / Мегапиксел. – Электрон. дан. – М., 2003. – Режим доступа : http:// homepage.corbina.net/~ksi/rout/st_rouz.htm. – Загл. с экрана.

22. Салюм, С. С. Разработка и исследование методов распознавания рукописных арабских текстов [Текст] : дис. … канд. техн. наук : 05.13.01 / С. С. Салюм. – Ижевск, 2003. – 127 с.

23. Системы безопасности: журнал для руководителей и специалистов в области безопасности.– М. : Leituvos rytas. 2009, №5 (89) .– 25000 экз.

24. Славин О.А. Вопросы распознавания текста, оцифрованного с помощью видеокамер [Электронный ресурс] / Г. О. Федоров; Cognitive Technologies. – Электрон. дан. – М., 2002. – Режим доступа : http://www.cognitive.ru/innovation/sbornic3/index.htm. – Загл. с экрана.

25. Славин, О. А. Древовидное распознавание нормализованных символов. [Текст] / О. А. Славин, А. А. Подрабинович // Интеллектуальные технологии ввода и обработки информации. Сборник трудов. – М.: Институт системного анализа РАН, 1998. – С. 137–157.

26. Черкас П.С., Веснин Е.Н. Автоматическое управление параметрами средств формирования изображений в системах распознавания автомобильных номеров реального времени [Текст] //

Технологии Microsoft в теории и практике программирования: материалы межвузовского конкурса-конференции студентов, аспирантов и молодых ученых Северо-Запада. – СПб. : Изд-во Политехн. ун-та, 2012. – С. 47– 48.

27. 32-bit Windows video surveillance software application that runs on Windows 95/98/2000/NT 4.0.: [Электронный ресурс]. М., 1997-2012. URL: http://www.gotchanow.com (Дата обращения: 18.02.2012).

28. ABBYY FineReader White Paper [Электронный ресурс] / Информационный портал DOCFLOW.ru. – Электрон. дан. – М., 2005. – Режим доступа : http://www.docflow.ru/analitics/detail.php?ID=15206. – Загл. с экрана.

29. ADVISOR specification documents: официальный текст, (internal classification 2001), pp. 437-806.

30. Almeida L., P. Pedreiras, J. Alberto, and G. Fonseca. The FFT-CAN protocol: why and how. IEEE Transactions on Industrial Electronics, 2002;49(6):1189–1201.

31. Arulampalam S., S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking. IEEE Transactions on Signal Processing, 2002;50(2):19–124.

32. Bennewitz M., W. Burgard, and S. Thrun. Using EM to learn motion behaviours of persons with mobile robots. In Proceedings of the International Conference on Intelligent Robots and Systems (IROS), Switzerland, 2002, pp. 502–507.

33. Black J., T. Ellis, and P. Rosin. A novel method for video tracking performance evaluation. In The Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, October, France, 2003, pp. 17–132, ISBN 0-7695-2022-7.

34. Boult T.E. ,R.J. Micheals, X. Gao, and M. Eckmann. Into the woods: visual surveillance of non-cooperative and camouflaged targets in complex outdoor settings. Proceedings of the IEEE, 2001;89(1):1314–1151.

35. Brodsky T., R. Cohen, E. Cohen-Solal et al. Visual surveillance in retail stores and in the home. In Advanced Video-Based Surveillance Systems. Kluwer Academic Publishers, Boston, 2001, pp. 50–20.

36. Christensen M. and R. Alblas. V2 – Design Issues in Distributed Video Surveillance Systems. Denmark, 2000, pp. 1–18.

37. Collins R.T., A.J. Lipton, T. Kanade et al. A System for Video Surveillance and Monitoring. Robotics Institute, Carnegie Mellon University, 2000, pp. 1–68. 87 C.S. Regazzoni, V. Ramesh, and G.L. Foresti. Special issue on video communications, processing, and understanding for third generation surveillance systems. Proceedings of the IEEE, 2001;89(10):1355–1365. Colorado, 1999, pp. 14–91, ISBN 0-7695-0037-4.

38. Conti M., L. Donatiello, and M. Furini. Design and analysis of RT-ring: a protocol for supporting real-time communications. IEEE Transactions on Industrial Electronics, 2002;49(6):1214–1226.

39. Decleir C., M.-S. Hacid, and J. Koulourndijan. A database approach for modelling and querying video data. In Proceedings of the 15th International Conference on Data Engineering, Australia, 1999, pp. 1–22.

40. Ferryman J.M., S.J. Maybank, and A.D.Worrall. Visual surveillance for moving vehicles. International Journal of Computer Vision, 37(2):187–143, Kluwer Academic Publishers, Netherlands, 2000.

41. Garcia L.M. and R.A. Grupen. Towards a real-time framework for visual monitoring tasks. In Third IEEE International Workshop

42. Gong S. and T. Xiang. Recognition of group activities using dynamic probabilistic networks. In Ninth IEEE International Conference on Compute Vision, France, 2003, vol.2, pp. 92–750, ISBN 0-7695-1950-4.

43. Greiffenhagen M., D. Comaniciu, H. Niemann, andV. Ramesh. Design, analysis, and engineering of video monitoring systems: an approach and a case study. Proceedings of the IEEE, 2001;89(10):1444–1517.

44. Gupta, Amarnath, and Ramesh Jain. "Managing event information: Modeling, retrieval, and applications." Synthesis Lectures on Data Management 3, no. 4 (2011): 1-141.

45. Hai Bui H., S. Venkatesh, and G.A.W. West. Tracking and surveillance in wide area spatial environments using the abstract hidden Markov model. IJPRAI, 2001;15(1):113–195.

46. Hemayed E.E.. A survey of self-camera calibration. In Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance, Florida, 2003, pp. 351–358, ISBN 0-7695-1431-7.

47. Impedovo, S. Frontiers in handwriting recognition [Text] / S. Impedovo // Fundamentals in handwriting recognition. – 1994. – Vol. 124. – P. 7–39.

48. Ivanov Y. and A. Bobick. Recognition of visual activities and interaction by stochastic parsing. IEEE Transactions of Pattern Recognition and Machine Intelligence, 2000;22(8):142–842.

49. Ivanov Y., C. Stauffer, A. Bobick, and W.E.L. Grimson. Video surveillance of interactions. In Second IEEE International Workshop on Visual Surveillance,

50. Jackson L.E. and G.N. Rouskas. Deterministic preemptive scheduling of realtime tasks. Computer, IEEE, 2002;35(5):42–5.

51. Jaynes C.,. Multi-view calibration from planar motion for video surveillance. In Second IEEE International Workshop on Visual Surveillance, Colorado, 1999, pp. 2–67, ISBN 0-7695-0037-4.

52. Kim, G. An architecture for handwritten text recognition systems [Text] / G. Kim, V. Govindaraju, S. N. Srihari // International Journal on Document Analysis and Recognition. – 1999. – Vol. 2. – № 1. – P. 37–44.

53. Kim, K. K. Learning-Based Approach for License Plate Recognition [Text] / K. K. Kim, K. I. Kim, J. B. H. Kim [and others] // IEEE Signal Processing Society Workshop, Neural Networks for Signal Processing Proceedings. – Australia: University of Sydney, 2000. – Vol. 2. – P. 614–623.

54. Maccario Carl Joseph. Aviation Security and. Nonverbal Behavior. Transportation Security Administration

55. Makris D., T. Ellis, and J. Black. Bridging the gaps between cameras. In International Conference on Multimedia and Expo, Taiwan, June 2004.

56. Marchesotti L., A. Messina, L. Marcenaro, and C.S. Regazzoni. A cooperative multisensor system for face detection in video surveillance applications. Acta Automatica Sinica, 2003;26(3):48–433.

57. Ng K.C., H. Ishiguro, M.Trivedi, andT. Sogo. Monitoring dynamically changing environments by ubiquitous vision system. In Second IEEE Workshop on Visual Surveillance, Colorado, 1999, pp. 67–9, ISBN 0-7695-0037-4.

58. Nguyen N.T., H.H. Bui, S. Venkatesh, and G.West. Recognising and monitoring high-level behaviour in complex spatial environments. In IEEE International Conference on Computer Vision and Pattern Recognition, Wisconsin, 2003, pp. 1–6.

59. Nguyen N.T., S. Venkatesh, G. West, and H.H. Bui. Multiple camera coordination in a surveillance system. Acta Automatica Sinica, 2003; 26(3):158–421.

60. Nishida, H. An approach to integration of off-line and on-line recognition of handwriting. [Text] / H. Nishida // Pattern Recognition Letters. – 1995. – Vol. 16. – № 11. – P. 2431–2438.

61. Niu, Wei, Jiao Long, Dan Han, and Yuan-Fang Wang. "Human activity detection and recognition for video surveillance." In Multimedia and Expo, 2004. ICME'04. 2004 IEEE International Conference on, vol. 1, pp. 719-722. IEEE, 2004.

62. Norhashimah P., H. Fang, and J. Jiang. Video extraction in compressed domain In IEEE Conference on Advanced Video and Signal Based Surveillance, Florida, 2003, pp. 321–327, ISBN 0-7695-1431-7.

63. Oren M., C. Papageorgiou, P. Sinham, E. Osuna, and T. Poggio. Pedestrian detection using wavelet templates. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Puerto Rico, 1463, pp. 193–199, ISSN 1016-6919/43.

64. Paulidis I. and V. Morellas. Two examples of indoor and outdoor surveillance systems. In Video-Based Surveillance Systems, P. Remagnino, G.A. Jones, N. Paragios, and C.S. Regazzoni, Eds. Kluwer Academic Publishers, Boston, 2002, pp. 39–51.

65. Pavlidis I., V. Morellas, P. Tsiamyrtzis, and S. Harp. Urban surveillance systems: from the laboratory to the commercial world. In Proceedings of the IEEE, 2001;89(10):1478–1495.

66. ProSystem CCTV : первый и единственный журнал в России по системам видеонаблюдения : профессиональное издание для экспертов и специалистов по охранному телевидению и видеонаблюдению .– М. : Немецкая Фабрика Печати. 2010, № 42-43. – 2000 экз.

67. Remagnino P., A. Baumberg, T. Grove, D. Hogg, T. Tan, A. Worral, and K. Baker. An integrated traffic and pedestrian model-based vision system. BMVC43 Proceedings, Israel, vol. 2, pp. 36–389, ISBN 0-9521-8449-5.

68. Rota N. and M. Thonnat. Video sequence interpretation for visual surveillance. In Third IEEE International Workshop on Visual Surveillance, Dublin, 2000, pp. 59–68, ISBN 0-7

69. Ryan N. Fries, Mostafa Reisi Gahrooei, Mashrur Chowdhury, Alison J. Conway, Meeting privacy challenges while advancing intelligent transportation systems, Transportation Research Part C: Emerging Technologies, Volume 25, December 2012, Pages 34-45, ISSN 0968-090X, http://dx.doi.org/10.1016/j.trc.2012.04.002.695-0644-4.

70. Rybski P.E., S.A. Stoeter, M. Gini, D.F. Hougen, and N.P. Papanikolopoulos. Performance of a distributed robotic system using shared communications channels. IEEE Transactions on Robotics and Automation, 2002; 18(5):713–427.

71. Saad A. and D. Smith. An IEEE 1346-firewire-based embedded video system for surveillance applications. In IEEE Conference on Advanced Video and Signal Based Surveillance, Florida, 2003, pp. 213–219, ISBN 0-7695-1431-7.

72. Stringa E. and C.S. Regazzoni. Content-based retrieval and real-time detection from video sequences acquired by surveillance systems. In International Conference on Image Processing, Chicago, 1464, pp. 138–142.

73. Tian, Ying-Li, Max Lu, and Arun Hampapur. "Robust and efficient foreground analysis for real-time video surveillance." In Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 1, pp. 1182-1187. IEEE, 2005.

74. Valera, M.; Velastin, S.A., "Intelligent distributed surveillance systems: a review," Vision, Image and Signal Processing, IEE Proceedings - , vol.152, no.2, pp.192, 204, 8 April 2005doi: 10.1049/ip-vis:20041147 Stauffer C., W. Eric, and L. Grimson. Learning patterns of activity using real-timetracking. IEEE Transact

75. Vesnin E. N., A. E. Mikhailov, V. A. Tsarev, and P. S. Cherkas. Smart Camera Adaptive Optoelectronic System [Text] // Pattern Recognition and Image Analysis. – Pleiades Publishing, Ltd., 2012. – 2012, Vol. 22, No. 3. – P. 406–411.

76. Wren C., A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder: real-time tracking of the human body. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1463;19(7):76–717.

77. Wu C.H., J.D. Irwin, and F.F. Dai. Enabling multimedia applications for factory automation. IEEE Transactions on Industrial Electronics, 2001;48(5):913–919.

78. Ye H., Gregory C. Walsh, and L.G. Bushnell. Real-time mixed-traffic wireless networks. IEEE Transactions on Industrial Electronics, 2001; 48(5):243–890.

79. Zhi-Hong Z.. Lane detection and car tracking on the highway. Acta Automatica Sinica, 2003;26(3):450–456.

Приложения

Приложение А. Исходный код модуля поддержания работоспособности сервера

```
1.  Public Class HealthMonitor
2.  Private _thread As New Threading.Thread(AddressOf HealthMonitorThread)
3.  Private Shared _lastTime As DateTime = Now
4.  Private _sl As ServiceLocator
5.  Public Property WatchdogTime As Integer = 120
6.  Public Sub New(sl As ServiceLocator)
7.  _sl = sl
8.  _thread.IsBackground = True
9.  _thread.Name = "HealthMonitor"
10. _thread.Priority = Threading.ThreadPriority.BelowNormal
11. _thread.Start()
12. End Sub
13. Public Shared Sub Ping()
14. _lastTime = Now
15. End Sub
16. Private Sub HealthMonitorThread()
17. Do
        a. Threading.Thread.Sleep(3000)
        b. Try
        c. Dim prc = Process.GetCurrentProcess
        d. Dim mem = (prc.WorkingSet64 / 1024 / 1024)
        e. Dim txt = "Память: " + mem.ToString("0.0") + ", ЦП: " +
prc.TotalProcessorTime.TotalMinutes.ToString("0.0")
        f. _sl.RootLogger.AddDebug(txt)
        g. If (mem > 1400) Then
            i. _sl.RootLogger.AddError("Потребление памяти слишком
        высоко. Вызываем GC.")
            ii. GC.Collect()
        h. End If
        i. Catch ex As Exception
        j. End Try
```

```
            k.      If (Now - _lastTime).TotalSeconds > WatchdogTime Then
            l.      Try
                i.      _sl.RootLogger.AddError("Слишком долгое бездействие.
            Выход.")
            m.      Catch ex As Exception
            n.      End Try
            o.      Try
                i.      _sl.KernelHost.Dispose()
            p.      Catch ex As Exception
            q.      End Try
            r.      Application.Exit()
            s.      Threading.Thread.Sleep(5000)
            t.      End
            u.      End If
        18. Loop
        19. End Sub
        20. End Class
```

Приложение В. Исходный код модуля взаимодействия с ядром распознавания

```
1.  Imports Imit.Systems.Config
2.  Imports Recar2.Core
3.  Imports Recar2.Algorithms
4.  Imports Imit.Imaging
5.
6.  Public Class FramesPair
7.      Public Property Work As IImageMatrix
8.      Public Property View As IImageMatrix
9.      Public Property Time As Date
10. End Class
11.
12.
13. Public Class KernelHost
14.     Implements IEventsSource
15.
16.     Private WithEvents _kernel As RecarKernel
17.     Private _kernelInitErrorText As String = ""
18.     Private _logger As Logger
19.     Private _buffer1 As New List(Of FramesPair)
20.     Private _buffer2 As New List(Of FramesPair)
21.     Private _storage As SettingsStorage
22.     Private _bufferSize As IntegerSetting
23.     Private _bufferStep As IntegerSetting
24.
25.     Public Event NewEvent(from As IEventsSource, data As EventDataPack)
Implements IEventsSource.NewEvent
26.     Public Event MotionDetected(sender As Object, e As MotionDetectedEventArgs)
27.     Public Event NumberRecognized(sender As Object, e As RecognizedEventArgs)
28.
29.     Public Event MatrixUpdated(sender As Object, e As
ChannelMatrixUpdatedEventArgs)
30.
31.     Public Sub InitKernel()
```

```
32.         If Init(False) = False Then
33.             If Init(True) = False Then
34.             End If
35.         End If
36.         Try
37.             Enabled = True
38.         Catch ex As Exception
39.         End Try
40.     End Sub
41.
42.     Public ReadOnly Property InitErrorText As String
43.         Get
44.             Return _kernelInitErrorText
45.         End Get
46.     End Property
47.
48.
49.     Private Function Init(client As Boolean) As Boolean
50.         _logger.AddMessage("Попытка инициализировать ядро...")
51.         Try
52.             _logger.AddInformation("Режим ядра: " + If(client, "ограниченный",
"полный"))
53.             _kernel = New RecarKernel(New Configuration(New XMLFileStorage()), 4)
54.             _kernel.IsClient = client
55.             _kernel.Init()
56.             _kernel.AdvancedMode = True
57.             _kernel.GetCamera(0).GetCurrentMatrixSnapshot()
58.             _logger.AddMessage("Ядро успешно инициализировано!")
59.             Return True
60.         Catch ex As Exception
61.             Try
62.                 _kernel.Dispose()
63.             Catch
64.             End Try
65.             _logger.AddWarning("Ядро не инициализировано:" + ex.Message)
66.             _kernelInitErrorText = ex.Message
67.             _kernel = Nothing
```

```
68.             Return False
69.         End Try
70.     End Function
71.
72.     Public ReadOnly Property Kernel As RecarKernel
73.         Get
74.             Return _kernel
75.         End Get
76.     End Property
77.
78.     Public ReadOnly Property IsWorking As Boolean
79.         Get
80.             Return _kernel IsNot Nothing
81.         End Get
82.     End Property
83.
84.     Private _isStarted As Boolean
85.
86.     Public Property Enabled As Boolean
87.         Get
88.             If IsWorking Then
89.                 Return _isStarted
90.             Else
91.                 Return False
92.             End If
93.         End Get
94.
95.         Set(value As Boolean)
96.             If IsWorking Then
97.                 If value <> _isStarted Then
98.                     If value = True Then
99.                         _kernel.Start()
100.                    Else
101.                        _kernel.Stop()
102.                    End If
103.                    _isStarted = value
104.                End If
```

```
105.            End If
106.          End Set
107.        End Property
108.
109.        Public ReadOnly Property IsLimited As Boolean
110.          Get
111.            If IsWorking Then
112.              Return _kernel.IsClient
113.            Else
114.              Return False
115.            End If
116.          End Get
117.        End Property
118.
119.        Private Sub _kernel_MatrixUpdated(sender As Object, e As
ChannelMatrixUpdatedEventArgs) Handles _kernel.MatrixUpdated
120.          HealthMonitor.Ping()
121.          RaiseEvent MatrixUpdated(sender, e)
122.        End Sub
123.
124.        Private Sub KernelMotionDetected(sender As Object, e As
MotionDetectedEventArgs) Handles _kernel.MotionDetected
125.          RaiseEvent MotionDetected(sender, e)
126.          If e.Camera.Id = 0 Then
127.            Static i1 As Integer
128.            i1 += 1
129.            If i1 >= _bufferStep.Value Then
130.              i1 = 0
131.              Dim frames As New FramesPair
132.              frames.Work = _kernel.GetCamera(0).GetCurrentMatrixSnapshot
133.              frames.View = (_kernel.GetCamera(1).GetCurrentMatrixSnapshot)
134.              frames.Time = Now
135.              _buffer1.Add(frames)
136.              Do While _buffer1.Count > _bufferSize.Value
137.                _buffer1.RemoveAt(0)
138.              Loop
139.            End If
```

```
        End If
        If e.Camera.Id = 2 Then
            Static i2 As Integer
            i2 += 1
            If i2 >= _bufferStep.Value Then
                i2 = 0
                Dim frames As New FramesPair
                frames.Work = _kernel.GetCamera(2).GetCurrentMatrixSnapshot
                frames.View = (_kernel.GetCamera(3).GetCurrentMatrixSnapshot)
                frames.Time = Now
                _buffer2.Add(frames)
                Do While _buffer2.Count > _bufferSize.Value
                    _buffer2.RemoveAt(0)
                Loop
            End If
        End If
    End Sub
```

Приложение С: Справка о внедрении системы

Открытое акционерное общество
"Ижорские заводы"
Ижорский завод д.б/н, Колпино, Санкт-Петербург, 196650
тел.:(812)322-80-00, факс:(812) 460-88-43
izhora@omzglobal.com www.omz-izhora.ru
ОКПО 05764417 / ОГРН 1027808749121 ИНН 7817005295 / КПП 783450001

Исх. 002/93
От «14» марта 2013г.

Советнику президента ОАО ОМЗ
А.И. Моргачу

**Отзыв на работу системы «Кербер»
на территории Ижорской промышленной площадки**

На Ижорской промышленной площадке в декабре 2012 года была установлена система контроля проезда автотранспорта «Кербер». В рамках данного проекта были установлены камеры и оборудование обработки и распознавания изображений на следующих КПП: «Фидерная», «КСМ», «36-й цех» (ООО «ОМЗ-Спецсталь»). Также был установлен сервер (в серверной ООО «ОМЗ-ИТ») и организован доступ к серверу из внутренней сети промплощадки и из сети Интернет.

В настоящее время система решает следующие задачи обеспечения безопасности: выполняет распознавание и фиксацию номеров с трех КПП, обеспечивает возможность их централизованного просмотра, их фильтрации и поиска, просмотра фотографий проехавших автомобилей, удаленного доступа к реестру событий, разграничения доступа к собранной системой информацией, контроля целостности работы системы, экспорта данных в xml (для интеграции с АСКВТ).

Система работает стабильно, распознает 98-100% (в зависимости от чистоты номерного знака) номеров проехавших автомобилей (в том числе и в ночное время). Имеет интуитивно понятный и удобный интерфейс, не требует настройки. Доступ к системе можно получить с любого мобильного или стационарного устройства с помощью логина и пароля.

Данная система может быть рекомендована для дальнейшего внедрения на всех объектах ОАО ОМЗ, имеющих проходные.

Генеральный директор И.А. Тимофеев