

Федеральное государственное бюджетное учреждение науки  
Институт автоматизации и процессов управления  
Дальневосточного отделения Российской академии наук

На правах рукописи



КИМ Александра Юрьевна

**РАСПОЗНАВАНИЕ ДИНАМИЧЕСКИХ ОБРАЗОВ  
ИНТЕЛЛЕКТУАЛЬНОЙ РАСПРЕДЕЛЁННОЙ  
ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНОЙ СИСТЕМОЙ  
СЕГМЕНТАРНОГО ТИПА**

Специальность: 05.13.18 – Математическое моделирование, численные  
методы и комплексы программ

Диссертация  
на соискание ученой степени  
кандидата технических наук

Научный руководитель:  
академик РАН, д.ф.-м.н.,  
профессор Ю.Н. Кульчин

Владивосток - 2015

*Посвящается моим родителям,  
Ким Юрию Михайловичу и Ким Анне Михайловне*

## ОГЛАВЛЕНИЕ

	Стр.
<b>ВВЕДЕНИЕ</b> .....	6
<b>1. СОВРЕМЕННОЕ СОСТОЯНИЕ ПРОБЛЕМЫ И АНАЛИЗ СУЩЕСТВУЮЩИХ МЕТОДОВ РАСПОЗНАВАНИЯ (КЛАССИФИКАЦИИ) ДИНАМИЧЕСКИХ ОБРАЗОВ</b> .....	14
1.1 ПРИНЦИПЫ ОРГАНИЗАЦИИ И СТРУКТУРА ИНТЕЛЛЕКТУАЛЬНОЙ РАСПРЕДЕЛЁННОЙ ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНОЙ СИСТЕМЫ.....	14
1.1.1 Структура ИРИИС на основе РВОИС сегментарного типа.....	19
1.2 МЕТОДЫ РАСПОЗНАВАНИЯ (КЛАССИФИКАЦИИ) ОБРАЗОВ.....	22
1.2.1 Методы факторного анализа.....	24
1.2.2 Методы регрессионного анализа.....	30
1.2.3 Нейронные сети.....	31
1.2.4 Методы поиска закономерностей (индукция правил).....	33
1.3 ВЫВОДЫ.....	37
<b>2. РАСПРЕДЕЛЁННАЯ ВОЛОКОННО-ОПТИЧЕСКАЯ ИЗМЕРИТЕЛЬНАЯ СЕТЬ СЕГМЕНТАРНОГО ТИПА</b> .....	39
2.1 ВИДЫ УКЛАДКИ ВОЛОКОННО-ОПТИЧЕСКИХ ЛИНИЙ В РАСПРЕДЕЛЁННОЙ ВОЛОКОННО-ОПТИЧЕСКОЙ ИЗМЕРИТЕЛЬНОЙ СЕТИ СЕГМЕНТАРНОГО ТИПА.....	40
2.1.1 Ранг матрицы измерительной системы, как первичный критерий качества укладки интегральных линий.....	43
2.1.2 Основные механизмы аналитического решения задачи реконструкции.....	44
2.1.3 Механизм сокращения числа неизвестных измерительной системы: реконструкция точечных распределений.....	44
2.1.4 Механизм увеличения числа уравнений измерительной системы: реконструкция гладких распределений.....	47
2.2 ФИЗИЧЕСКИЕ ПРИНЦИПЫ ФУНКЦИОНИРОВАНИЯ РАСПРЕДЕЛЁННОЙ ВОЛОКОННО-ОПТИЧЕСКОЙ СЕТИ СЕГМЕНТАРНОГО ТИПА.....	52
2.2.1 Общие принципы построения интегрирующих волоконно-оптических измерительных линий.....	52
2.2.2 Волоконно-оптические датчики на основе изгиба оптоволокна.....	54
2.2.3 Волоконно-оптические датчики на основе одноволоконного многомодового интерферометра.....	63
2.2.4 Модель конструкции чувствительного элемента для волоконно-оптической измерительной линии на основе ОМИ.....	65

2.3	КОРРЕЛЯЦИОННАЯ ОБРАБОТКА СИГНАЛА НА ВЫХОДЕ ОДНОВОЛОКОННОГО МНОГОМОДОВОГО ИНТЕРФЕРОМЕТРА ПРИ МАКРОИЗГИБЕ.....	67
2.3.1	Обработка выходного оптического сигнала с чувствительного элемента на основе ОМИ.....	69
2.4	РАЗРАБОТКА ВОЛОКОННО-ОПТИЧЕСКОЙ ИЗМЕРИТЕЛЬНОЙ СИСТЕМЫ ОХРАНЫ ПЕРИМЕТРА.....	77
2.4.1	Принципы создания чувствительного элемента на основе многомодового волоконного световода.....	78
2.4.2	Принципы создания чувствительного элемента на основе одномодового волоконного световода.....	84
2.4.3	Создание волоконно-оптической измерительной линии на основе одномодового волоконного световода.....	88
2.4.4	Макет волоконно-оптической измерительной сети системы охраны периметра.....	94
2.5	ВЫВОДЫ.....	98
<b>3.</b>	<b>СИСТЕМА ОБРАБОТКИ ДАННЫХ РАСПРЕДЕЛЁННОЙ ВОЛОКОННО- ОПТИЧЕСКОЙ СЕТИ СЕГМЕНТАРНОГО ТИПА.....</b>	<b>100</b>
3.1	СИСТЕМА ОБРАБОТКИ ДАННЫХ ДЛЯ РАСПОЗНАВАНИЯ ДИНАМИЧЕСКОГО ОБРАЗА С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ.....	103
3.1.1	Разработка генератора выборки примеров сенсорных данных.....	103
3.1.2	Способы сжатия данных для формирования моделей образов по сгенерированным данным.....	113
3.1.3	Модуль преобразования реальных данных с макета ВОИС охраны периметра.....	119
3.1.4	Способы сжатия данных для формирования модели образа по реальным данным с макета ВОИС охраны периметра.....	128
3.1.5	Результаты численного моделирования и обработки по сгенерированным данным с применением нейронных сетей.....	133
3.1.6	Результаты исследований при обработке реальных данных с макета ВОИС охраны периметра с применением нейронных сетей.....	151
3.2	ОБРАБОТКА ДАННЫХ ДЛЯ РАСПОЗНАВАНИЯ ДИНАМИЧЕСКОГО ОБРАЗА С ИСПОЛЬЗОВАНИЕМ ПРОДУКЦИОННОЙ НЕЧЁТКОЙ СИСТЕМЫ.....	156
3.2.1	Реализация продукционной нечёткой системы в виде нечёткого дерева решений по реальным данным с макета ВОИС охраны периметра.....	158
3.2.2	Результаты обработки данных продукционной нечёткой системой в виде нечёткого дерева решений.....	164
3.3	ВЫВОДЫ.....	172
	<b>ЗАКЛЮЧЕНИЕ.....</b>	<b>173</b>
	<b>СПИСОК ЛИТЕРАТУРЫ.....</b>	<b>175</b>

<b>СПИСОК СОКРАЩЕНИЙ</b> .....	194
Приложение № 1.....	195
Приложение № 2.....	201
Приложение № 3.....	208
Приложение № 4.....	214
Приложение № 5.....	215

## ВВЕДЕНИЕ

Интеллектуальную интеграцию и развитие современных технологий в настоящий момент невозможно представить без расширения научных исследований в области волоконно-оптических информационно-измерительных систем для распределённого мониторинга быстропротекающих явлений, происходящих в широком диапазоне изменения температуры и давления, а также других факторов влияния окружающей среды. В связи с этим, особое значение приобретает применение распределённых волоконно-оптических измерительных сетей (РВОИС) [1], способных не только обеспечить мониторинг состояния масштабных и многопараметрических систем и объектов в реальном времени, но и адаптироваться под конкретные воздействия окружающей среды. В таких сетях используют распределённые волоконно-оптические преобразователи, на основе которых измерения производятся непрерывно вдоль траектории их размещения. Принципиальное отличие распределённых волоконно-оптических преобразователей по технологии их изготовления состоит в использовании для них специальной регистрирующей техники, позволяющей разделять результаты измерения параметров физических полей по длине оптоволокна [2]. Поэтому разработка новых подходов обработки и анализа получаемых от измерительных линий распределённой волоконно-оптической измерительной сети больших массивов данных будет полезна для решения целого ряда перспективных задач в области охранных комплексов потенциально опасных объектов, авиа- и космических, морских и автомобильных конструкций, мониторинга гидротехнических и строительных сооружений, а также при создании автоматизированных систем контроля и управления техническими процессами и объектами.

При этом для РВОИС следует решать следующие задачи: восстановление пространственного распределения параметров физического поля и выявление его структурных неоднородностей в режиме реального времени. Тем самым

воссоздаётся динамическая картина воздействия на РВОИС. Затем, по совокупности всей информации необходимо выделить характерные признаки и дать прогноз развития ситуации или распознать объект (совокупность объектов). Задача восстановления функции распределения параметров физического поля, которые регистрируются РВОИС, непосредственно связана с решением проблемы обеспечения стабильности рабочих характеристик очень большого числа волоконно-оптических датчиков и измерительных линий, которые необходимо регулярно размещать в пространстве, что практически неосуществимо в реальных условиях. В итоге, мы имеем ряд проблем, которые приводят к снижению метрологических характеристик РВОИС и обуславливают необходимость анализа большого объёма объективной и субъективной информации, связанной с плохо формализуемыми задачами различной физической природы. Введение способности к обучению при обработке информационных массивов данных, полученных от РВОИС, достигаемой через внедрение в их состав нейронных сетей, позволяет адаптировать вычислительный процесс к реализуемой на практике ситуации [3]. Все выше перечисленные факторы стали основными аргументами для проведения работ, направленных на использование современных научных подходов, таких как прикладная статистика и методы анализа данных в сочетании с нейросетевыми технологиями и нечёткой логикой при разработке системы обработки информации (СОИ) для РВОИС [4].

На современном уровне процесс мониторинга объектов с помощью распределённых измерительных систем (РИС) не сводится лишь к обнаружению самого объекта, а требует оценки и анализа полученных данных, регистрации и воспроизведения важнейших характеристик состояния объекта в реальном времени, передачи обработанной информации лицу, принимающему решения. Это новое понимание мониторинга как информационного мониторинга сейчас получило развитие и делает более ёмким его по содержанию. Первые публикации работ исследователей по распознаванию динамических образов и идентификации объектов появились в США, Швеции, Франции, Японии в начале 1980-х гг. В России в данной области хорошо известны работы таких выдающихся учёных,

как академика РАН, д.ф.-м.н Ю. И. Журавлёва, член-корреспондента РАН, д.т.н. В. А. Сойфера, д.т.н. Л. М. Местецкого, д.т.н. Н. Г. Загоруйко, д.т.н. В. П. Пяткина и др. Современные системы мониторинга оснащены блоками поддержки принятия решений. Они предоставляют пользователю не только конкретные параметры, но и дают их интерпретацию в различных возможных кризисных ситуациях, что помогает принимать адекватные быстрые решения в чрезвычайных случаях. В связи с многообразием функций и требований для таких систем, многие проблемы при распознавании динамических образов остаются нерешёнными, а при моделировании и обработке данных необходимы более эффективные подходы, методы и алгоритмы.

При мониторинге движущихся объектов для распознавания динамических образов целесообразен комплексный подход решения проблем обработки данных, который и был реализован в данной работе с помощью создания информационно-измерительной системы (ИИС) на основе распределённой волоконно-оптической измерительной сети, что повышает как функциональность, так и надёжность, скорость и эффективность процессов обработки в созданной системе.

Разработка общего подхода к созданию нового класса ИИС как интеллектуальной распределённой информационно-измерительной системы (ИРИИС) на основе РВОИС сегментарного типа с применением «гибких технологий» таких, как экспертные системы (ЭС), нейронные сети (НС), продукционные нечёткие системы (ПНС) и гибридные системы (ГС) позволит расширить область охвата решаемых задач с привлечением экспертных знаний.

Объектом исследования являются подходы проектирования и функционирования ИРИИС на основе РВОИС сегментарного типа. Предметом исследования являются подходы, способы и методы обработки информации в ИРИИС по данным, полученным с помощью РВОИС, моделирование динамических образов распознавания, критерии качества функционирования этой системы.

**Целью** диссертационной работы является моделирование и исследование подходов, способов и методов распознавания динамических образов и создание



программно-алгоритмического комплекса ИРИИС на основе РВОИС сегментарного типа.

Для достижения поставленной цели в работе требуется решить следующие **задачи**:

- провести анализ существующих ИИС, по результатам которого определить требования к принципам построения и структуре создаваемой ИРИИС на основе РВОИС и разработать критерии её функционирования;
- разработать основные подходы к построению ИРИИС на основе РВОИС, позволяющие формировать топологию такой системы, составные её блоки и модули, алгоритмы функционирования системы;
- разработать и исследовать способы и методы обработки сгенерированных данных РВОИС для распознавания динамических образов и взять их за основу для проектных решений при создании ИРИИС;
- разработать и исследовать действующий макет волоконно-оптической измерительной сети (ВОИС) охраны периметра;
- проанализировать, протестировать работу ИРИИС на данных с макета ВОИС охраны периметра для подтверждения правильности исходных положений, предлагаемых способов, методов обработки и созданных алгоритмов;
- разработать методологическую базу и программно-алгоритмический комплекс для ИРИИС на основе РВОИС.

**Научная новизна** диссертационной работы определяется следующими результатами:

1. Разработаны обобщенная, функциональная и структурная модели ИРИИС на основе РВОИС сегментарного типа для распознавания динамических образов в зоне мониторинга измерительной сети.
2. Предложен и обоснован новый подход к решению задачи идентификации известных классов объектов с изменяющимися параметрами по сигналам измерительной сети при построении ИРИИС на основе РВОИС сегментарного типа.

3. Предложен и алгоритмически реализован набор специальных способов и методов обработки данных в ИРИИС на основе РВОИС сегментарного типа для идентификации движущихся объектов в зоне мониторинга измерительной сети.

***Теоретическая и практическая значимость.*** Результаты работы могут найти применение при разработке нового класса ЭС для решения сложных плохо формализуемых прикладных проблем, требующих знаний экспертов. Предложенные в диссертации способы, методы и комплекс программ обработки данных РВОИС для идентификации движущихся объектов, а также разработанные программные модули (генератор сенсорных данных РИС и модуль преобразования данных) применимы в различных сферах деятельности, связанных с обработкой больших массивов данных при распознавании динамических образов.

***Методы исследования.*** При решении поставленных в работе задач использовались методы системного анализа, математического моделирования и прикладной статистики, искусственного интеллекта, разработки информационных и экспертных систем, нечёткая логика, а также методы программирования.

***Положения, выносимые на защиту:***

1. Обобщенная, функциональная и структурная модели ИРИИС на основе РВОИС сегментарного типа для распознавания динамических образов в зоне мониторинга измерительной сети.
2. Новый подход к решению задачи идентификации известных классов объектов с изменяющимися параметрами по сигналам, полученным при переменном воздействии на измерительную сеть, для построения ИРИИС на основе РВОИС сегментарного типа.
3. Набор специальных способов и методов обработки данных в ИРИИС на основе РВОИС сегментарного типа для идентификации движущихся объектов в зоне мониторинга с применением моделирования динамических образов и их реализация в виде продукционной нечёткой системы и экспертной системы с привлечением нейросетевых технологий.

**Реализация результатов работы.** Экспериментальная проверка используемых в работе подходов, методов, способов и алгоритмов обработки данных ИРИИС проведена для данных, полученных с созданного действующего макета ВОИС охраны периметра.

**Обоснованность и достоверность** приведенных результатов работы обеспечиваются корректным использованием теории моделирования, методов теории машинного обучения, проектирования информационных систем и системного подхода.

**Апробация** основных положений диссертационной работы проведена на конференциях: НТК «Молодёжь и научно-технический прогресс» - Владивосток, 2004; Asia-Pacific Conference on Fundamental Problems of Opto- and Microelectronics – APCOM’07, APCOM’09 (Vladivostok, 2007, 2009), APCOM’14 (Tokyo, 2014, Japan); Всероссийская НТК «Нейроинформатика 2009» – Москва, 2009; 20th Anniversary Korea-Russia Science Conference – Baikal, 2010, Russia; Fundamental Science and Technology – Promising Developments IV – North Charleston, 2014, USA.

**Публикации.** По теме диссертационной работы опубликованы 14 работ, из которых 4 – в изданиях из списка ВАК.

**Личный вклад автора.** Постановка задачи исследований осуществлена совместно с научным руководителем. Основные результаты диссертации получены лично автором. Соискатель участвовала в постановке задач, разработке и обосновании методов их решения, интерпретации результатов. Автором разработаны, проверены и протестированы компьютерные программы для реализации предложенных в диссертации подходов, способов и методов.

**Структура и объем диссертации.** Диссертация состоит из введения, трёх глав, заключения, списка используемой литературы и приложений. Общий объем диссертации составляет 223 страниц, основной текст изложен на 194 страницах. Работа содержит 77 рисунков и 10 таблиц.

В **первой главе** представлен обзор по состоянию решения проблемы распознавания образов, как с помощью традиционных методов, так и с помощью

систем искусственного интеллекта с использованием нейронных сетей и нечёткой логики. Рассмотрена структура ИРИИС на основе РВОИС сегментарного типа, принципы её организации и функционирования.

В конце первой главы формализуется постановка основной задачи исследований диссертационной работы, формулируются выводы по первой главе, в которых отражаются основные результаты главы.

**Вторая глава** посвящена решению задач: оптимизации укладки измерительных линий сети, разработке подходов построения РВОИС сегментарного типа, созданию ВОИС охраны периметра и методам обработки сигналов. По основным результатам второй главы формулируются выводы.

В **третьей главе** проводится моделирование и исследование основных процессов реализации ИРИИС с учётом особенностей применения их для распознавания движущихся объектов и их параметров, по сгенерированным данным с помощью генератора выборки примеров сенсорных данных и измеренным данным с макета ВОИС охраны периметра, сделаны оценки эффективности функционирования такой системы. А также исследованы и протестированы подходы, методы и способы, применяемые при проектировании модуля – системы обработки данных ИРИИС. Для этого в рамках разработанных подходов к созданию ИРИИС используются моделирование, нейронные сети, процесс нечёткого вывода и другие средства построения моделей систем. В конце главы подведены итоги и определены основные результаты.

В **заключении** приведены основные результаты и выводы, имеющие научную и практическую ценность.

**Благодарности.** Автор признателен и выражает глубокую благодарность своему научному руководителю – академику РАН, д.ф.-м. н., профессору Юрию Николаевичу Кульчину за мудрое руководство, терпение, участие и готовность к обсуждению тонкостей исследовательской работы по теме диссертации, полезные советы и замечания. Также автор благодарит за содействие при воплощении идей к.т.н., с.н.с. ИАПУ ДВО РАН Бориса Сергеевича Ноткина. За поддержку,

корректные рекомендации и замечания по материалам диссертации автор выражает признательность к.т.н. Гамаюнову Е.Л., д.ф.-м.н. Дзюбе В.П. и другим сотрудникам лаборатории прецизионных оптических методов измерения (№21) научного отдела оптоэлектронных методов исследования газообразных и конденсированных сред (№20) ИАПУ ДВО РАН.

# **1. СОВРЕМЕННОЕ СОСТОЯНИЕ ПРОБЛЕМЫ И АНАЛИЗ СУЩЕСТВУЮЩИХ МЕТОДОВ РАСПОЗНАВАНИЯ (КЛАССИФИКАЦИИ) ДИНАМИЧЕСКИХ ОБРАЗОВ**

Дан анализ современного состояния проблемы распознавания образов, использующие традиционные методы и методы, основанные на применении нейронных сетей и нечёткой логики.

## **1.1 ПРИНЦИПЫ ОРГАНИЗАЦИИ И СТРУКТУРА ИНТЕЛЛЕКТУАЛЬНОЙ РАСПРЕДЕЛЁННОЙ ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНОЙ СИСТЕМЫ**

Развитие и совершенствование современных информационно-измерительных систем при неуклонном расширении диапазона их потенциального применения приобретает огромное значение. В последние годы появилась необходимость поиска новых подходов построения и организации информационно-измерительных систем, требующих исследованных физических, технических и технологических решений. Подобные системы должны быть способны обеспечивать надежное и достоверное представление о контролируемом и возможно движущемся объекте, о его состоянии при воздействии случайных возмущений, при нечеткости поступающей информации и т. д. [1-17]. В настоящее время эта сложная задача распознавания динамических образов не имеет решения в общем виде, однако в случае ряда прикладных задач для этого могут стать полезными и достаточно эффективными исследования подходов на основе комплексного привлечения методов и технологий искусственного интеллекта. Поэтому обоснован и необходим поиск принципиально новых подходов решения этой проблемы, которые можно реализовать с помощью создания интеллектуальных распределенных информационно-измерительных систем.

Сфера потенциальных применений таких интеллектуальных распределенных информационно-измерительных систем охватывает огромный

спектр различных прикладных задач. К ним относятся проблемы контроля состояния и управления эксплуатируемыми в экстремальных условиях разнообразными крупномасштабными объектами (например, такими как мосты, тоннели, высотные здания, крупные социально-культурные и спортивные сооружения, нефтяные платформы, нефте- и газопроводы, самолеты, корабли, космические аппараты и т.д.), направленных на предотвращение техногенных аварий,

Внедрение в практику ИРИИС требует создания и использования разнообразных измерительных устройств и систем, позволяющих получать в реальном времени достоверную картину о состоянии контролируемых объектов и о протекающих процессах, оснащения их мощными вычислительными и программными средствами, что позволяет выполнять сложные процедуры измерений. В результате возникает необходимость создания новых подходов к самому измерительному процессу, а также к выбору измерительного оборудования. Как следствие этого, возникает потребность развития новой идеологии и поиска новых системных решений при разработке и эксплуатации измерительных устройств, отвечающих современным требованиям. Поэтому, если ранее многие физические величины измерялись независимо друг от друга, отдельными измерительными приборами в отдельных точках, теперь четко обозначились тенденции их одновременного и интегрированного сравнения, установления их распределения и закономерностей взаимодействия [1-8].

Значительный прогресс техники волоконно-оптических линий связи (ВОЛС) у нас в стране и за рубежом привел к возникновению нового направления метрологии – волоконно-оптические датчики (ВОД) физических величин [23-25]. Сочетание развитой технологии ВОЛС с современными достижениями в областях опто- и микроэлектроники открывают широкие перспективы для создания разнообразных датчиков физических величин [1]. При этом важными достоинствами ВОД являются высокая чувствительность, малые размеры, устойчивость к электромагнитным помехам и агрессивным воздействиям окружающей среды, возможность мультиплексирования отдельных датчиков в

сложные измерительные системы и потенциально низкая стоимость. Поэтому применение волоконно-оптической элементной базы открывает широкие перспективы для создания помехозащищенных измерительных сетей, способных осуществлять сбор измерительной информации о состоянии различных контролируемых объектов. Как показано в [1-6], значительный фундаментальный и практический интерес для решения проблемы создания высокочувствительных распределенных измерительных систем представляют волоконные измерительные интерферометры, которые способны эффективно интегрироваться с оптоэлектронными и компьютерными нейронными вычислительными сетями для обработки получаемой информации. Важной составной частью данных РИС являются распределенные волоконно-оптические измерительные сети, основанные на применении волоконных интерферометров (ВИ) в качестве протяженных измерительных линий, выполняющих задачу измерения определенных физических параметров контролируемых объектов. Принцип действия таких РВОИС базируется на использовании томографического подхода [11-22] в их структурной организации и реконструкции пространственной функции распределения измеряемых физических параметров, а также на применении нейронных сетей для обработки получаемых данных [1, 9, 10, 19-22]. Основное преимущество нейросетевого подхода [40-49] – возможность выявления закономерностей в данных, их обобщение, т.е. извлечение знаний из данных, а основной недостаток – невозможность непосредственно (в явном виде, а не в виде вектора весовых коэффициентов межнейронных связей) представить функциональную зависимость между входом и выходом для состояний исследуемого объекта. Недостатком нейросетевого подхода является трудность формирования представительной выборки, большое число циклов обучения и забывание «старых» примеров, трудность определения размера и структуры сети, но это вполне преодолимо в ИРИИС, которую мы реализовали. При использовании РВОИС характеристики распространяющегося вдоль оптического волокна ВИ излучения изменяются в зависимости от характеристик воздействующих физических полей, а сигнал на выходе измерительной линии



представляет собой линейный интеграл от функции распределения исследуемого параметра физического поля. Математически проблема может трактоваться как классическая томографическая задача, так как для исследуемых полей количество информационных каналов меньше, чем число требуемых решений [14-16]. В связи с чем, для обработки, поступающей от РВОИС информации, и решения томографической задачи в таком случае было целесообразно применение нейронных сетей, поскольку их отличительной особенностью является возможность обучения и адаптации к конкретным условиям, а также высокая скорость обработки данных [27-40].

Использование ИРИИС для мониторинга состояний объектов, регистрируемых РВОИС, сопряжено с решением проблемы классификации поступающих данных об объектах и автоматического принятия на их основе в режиме реального времени решения о дальнейших действиях. Для решения данной проблемы также весьма перспективно применение технологий, основанных на использовании нейронных сетей.

Проблема восстановления и классификации образов в нейросетевом базисе описана в целом ряде работ [34-40]. Результаты таких исследований находят применение в различных отраслях науки и техники (в медицине, робототехнике, географии и астрофизике), но у разработанных методов есть существенный недостаток, они основываются на статичности или квазистатичности самих образов и параметров окружающей среды, что существенно ограничивает возможности применения данных подходов в случае решения динамических задач.

Как показали результаты исследований [1, 34-54], использование РИС и нейронных сетей для обработки сигналов РВОИС выдвигает на передний план ряд первоочередных проблем:

- необходимость решения чрезвычайно трудоёмкой задачи обучения нейронной сети, требующей применения оптимизационных алгоритмов. В результате, в процессе обучения нейронной сети возникает задача минимизации функции

ошибки рассогласования относительно большого количества (порядка нескольких тысяч) весовых коэффициентов матриц связи. Кроме этого профиль многомерной поверхности минимизируемой функции в пространстве её переменных имеет очень сложный характер и, как следствие, данная задача требует длительного времени и больших вычислительных мощностей для ее решения;

- поступающая информация с РИС определяется не только воздействием физического поля контролируемого объекта. Она зависит от разброса параметров измерительных линий, наличия дефектов в топологии реальных протяженных РВОИС, вызванных отклонением профиля распределения от идеального, шума, возникающего из-за неконтролируемых воздействий со стороны окружающей среды. Всё указанное приводит к ошибкам при распознавании и чрезвычайно трудно учитывается при обучении нейронной сети;

- при контроле движущихся объектов изменчивость картины данных, полученной с помощью РИС, затрудняет, а в ряде случаев делает невозможным оперативное выделение признаков, по которым его можно классифицировать.

Для решения выше обозначенных проблем было предложено разбить РВОИС на части и представить сеть в виде большого числа одинаковых и самостоятельных измерительных сегментов (ИС) столь малой величины, чтобы разброс параметров измерительных линий и дефекты топологии всей РИС не вызывали существенного отклонения характеристик ИС друг от друга. При такой структуре РВОИС появилась возможность произвести унификацию обучения нейронных сетей для всех ИС и снизить время обучения НС. Такой подход, при организации конвейерного принципа опроса ИС позволяет вообще унифицировать модуль обработки данных (МОД) для всех ИС. В этом случае процесс распознавания образа ИРИИС на основе РВОИС включает несколько стадий. Сначала по информации, полученной от ИС, осуществляют быстрый поиск представляющих интерес образов безотносительно к конкретному его местоположению (т.е. ИРИИС работает в ждущем режиме). Если интересующие нас образы будут найдены, то на следующем шаге производится выделение ИС,

содержащих эти образы, определяется их расположение, а также другие характеристики позволяющие отнести их к известному классу образов.

Настоящая работа представляет разработанные принципы организации и построения ИРИИС на основе РВОИС большой размерности сегментарного типа, обеспечивающие при достаточно хорошем качестве распознавания образов высокую скорость обработки данных для принятия управленческих решений в реальном времени [50, 51].

### **1.1.1 Структура ИРИИС на основе РВОИС сегментарного типа**

Предлагаемая в работе структура ИРИИС состоит из блоков и модулей, организованных таким образом, что между ними распределены все функции, выполняемые системой в целом [50].

В процессе работы ИРИИС осуществляет следующие функции:

а) через систему волоконно-оптических (ВО) разветвителей модули лазерных излучателей возбуждают соответствующие измерительные линии сегментов РВОИС;

б) световое излучение, выходящее из измерительных линий сети, принимают фотоприёмники, а затем сигналы поступают в формирователь массива выходных сигналов и аналого-цифровой преобразователь (АЦП);

в) из полученных на выходе данных РВОИС, представленных в виде блочной матрицы с помощью её преобразования, формируют блочную матрицу признаков образа, в дальнейшем мы её и будем называть моделью образа (МО);

г) массив данных МО подают на вход многоканальной нейронной сети (МНС);

д) определение МНС принадлежности МО к заданным классам образов позволяет перейти к принятию управляющих решений системой.

Обобщённая схема ИРИИС показана на рисунке 1.1.

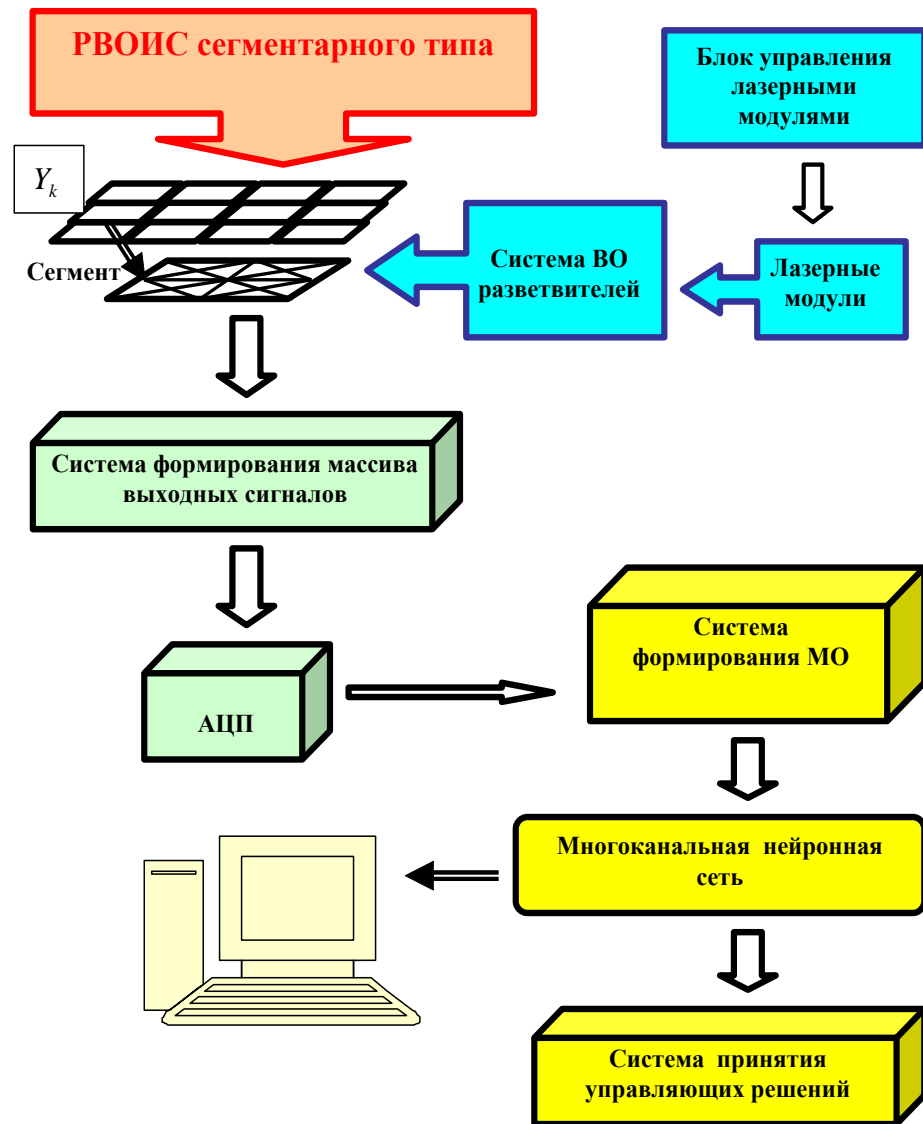


Рис.1.1. Схема ИРИИС на основе РВОИС сегментарного типа

Особенностью РВОИС является её сегментарное строение. Каждый отдельный ИС состоит из набора линий с датчиками и имеет определённую заданную конфигурацию. Сегменты РВОИС одинаковы, поэтому каждому из них присваивается цифровой код. Как было отмечено выше, искажениями, которые создаёт ИС, можно пренебречь из-за их малости. Сбор информации в РВОИС происходит последовательно от сегмента к сегменту в заданной очередности и циклично. По данным опроса РВОИС формируется блочная матрица. Формирование матрицы завершается после нескольких циклов опроса РИС. Количество циклов опроса зависит от порядка динамического образа, а размеры

блока матрицы соответствуют числу сегментов в РВОИС. Алгоритм обработки данных ИС аналогичен для всех сегментов, поэтому после преобразования блочной матрицы в МО дальнейшая обработка в ИС при классификации образа может быть распараллелена, а обучение ИС можно проводить на выборке данных для одного из ИС.

Таким образом, функционирование таких систем как ИРИИС будет складываться из выполнения следующих этапов:

I этап. Регистрация данных РВОИС и получение модели реального образа (МРО).

II этап. Сравнение МРО с имеющимися образами в памяти ИРИИС и определение ИС, в котором интересующий образ найден, с определением принадлежности данного образа к одному из известных классов образов.

III этап. Принятие решения и, при необходимости, выработка рекомендаций для проведения дальнейшего распознавания с поиском дополнительных признаков, характеризующих образ и его поведение в сегменте.

Ключевым модулем ИРИИС, который потребовалось тщательно проработать для реализации выше перечисленных этапов, стала система обработки данных, обеспечивающая эффективность решения класса сложно формализуемой проблемы, а также гибкость и скорость принятия решений при распознавании динамического образа (в нашем случае рассматривалась задача идентификации различных движущихся объектов). Наш модуль обработки данных представляет собой алгоритмический комплекс, реализуемый в виде системы классификации образов по типу экспертных систем.

В наши дни системы классификации должны рассматриваться в более широком смысле, если мы хотим решать задачи классификации более сложной и интеллектуальной природы. Первый уровень этой системы получает сенсорные данные, генерируемые некоторым источником информации. Второй уровень извлекает множество признаков, характеризующих полученные сенсорные данные. Третий уровень классифицирует эти признаки в одну или несколько различных категорий, которые затем помещаются в глобальный контекст на

четвёртом уровне. В заключение можно для примера поместить разобранный входной сигнал в некоторую форму базы данных, которую будет использовать конечный пользователь [45]. В процессе реализации этой системы могут использоваться различные методы классификации образов и подходы построения интеллектуальных систем, между которыми на практике нет чёткой границы, т.к. очень часто для решения поставленных задач используют смешанные системы [53, 54].

Выполненная работа показывает целесообразность комплексного подхода к решению проблемы распознавания динамических образов (на примере идентификации движущихся объектов в зоне мониторинга РВОИС). Таким образом, комбинирование нейронных сетей в сочетании с другими соответствующими средствами позволяет обеспечить следующие качества ИРИИС:

- **Робастность** системы характеризуется устойчивостью и работоспособностью при распознавании зашумлённого или неполного входного сигнала; повреждённая ИРИИС может продолжать выполнять свои функции на удовлетворительном уровне, а обучение не обязательно должно быть совершенным.
- **Гибкость** системы определена концепцией обработки информации в нейронных сетях, которая проистекает из принципа параллелизма.

## 1.2 МЕТОДЫ РАСПОЗНАВАНИЯ (КЛАССИФИКАЦИИ) ОБРАЗОВ

Выбор комплексного подхода, реализуемый в рамках решения проблемы распознавания (классификации) динамических образов в ИРИИС требует поиска наиболее целесообразного сочетания методов на основе **теории вычислительного обучения** для построения алгоритмов, способных обучаться по прецедентам. Эти методы тесно связаны с **извлечением информации и интеллектуальным анализом данных**, а их практическая работоспособность была подтверждена экспериментами на модельных и реальных данных.

Предобработка, фильтрация и сжатие данных, а также сокращение размерности признаков, которыми описывается МО, с помощью методов *прикладной статистики* позволяет избавиться от шумов и ошибок в данных, которые могут вызвать неточность при настройке модели восстанавливаемой зависимости по обучающей выборке. Это повлияет на значение функционала качества, а также отразится на качестве при классификации динамических образов в ИРИИС. Алгоритм обучения ищет такой набор параметров модели, при котором функционал качества на заданной обучающей выборке принимает оптимальное значение, а сам процесс *настройки* модели по выборке данных сводится к применению *численных методов оптимизации*.

Методология применения методов *прикладной статистики* основывается на предположении о вероятностной интерпретации анализируемой информации и получении в результате применения этих методов закономерностей, имеющих стохастический характер. Информация, которую подвергают анализу и обработке в ИРИИС именно такого рода, а поэтому выбор методов прикладной статистики представляется наиболее приемлемым и приоритетным для выполнения преобразований и предобработки данных в этой системе.

Разные подходы к задаче обучения могут привести к одной и той же модели, но разным методам обучения. Существует целый ряд хорошо известных подходов классификации – это *статистическая классификация; классификация на основе сходства; классификация на основе разделимости; нейронные сети; индукция правил (поиск закономерностей); кластеризация; регрессия; алгоритмические композиции; сокращение размерности; выбор модели; байесовский вывод*. Разделение методов «по подходам» довольно условно. Анализ этих методов и проблемы, которые было необходимо решать при распознавании динамических образов, привели к выбору тех из них, которые больше соответствуют предназначению ИРИИС. На отдельных интересных аспектах, используемых в работе подходов и методов, хотелось бы остановиться подробнее в этом разделе главы.

### 1.2.1 Методы факторного анализа

Одним из основных подходов анализа данных, используемых в ИРИИС для сжатия данных и формирования МО, выполняется на основе методов факторного анализа [55, 56].

**Факторный анализ** – метод многомерного статистического анализа, позволяющий на основе экспериментального наблюдения признаков объекта выделить группу переменных, определяющих корреляционную взаимосвязь между признаками. Суть факторного анализа состоит в том, что на основе исследования корреляционных взаимосвязей признаков находить причины, определяющие эти взаимосвязи. В общем случае моделью описываемой взаимосвязи является набор линейных уравнений, коэффициентами этих уравнений являются нагрузки, которые показывают «вес» каждого из факторов для данного признака. Факторные нагрузки – это значения коэффициентов корреляции каждого из исходных признаков с каждым из выявленных факторов. Чем теснее связь данного признака с рассматриваемым фактором, тем выше значение факторной нагрузки. Положительный знак факторной нагрузки указывает на прямую связь данного признака с фактором, а отрицательный знак – на обратную связь. Факторными весами называют количественные значения выделенных факторов для каждого из имеющихся объектов. Объекту с большим значением факторного веса присуща большая степень проявления свойств, определяемых данным фактором. Для большинства методов факторного анализа факторы определяют как стандартизованные показатели с нулевой средней и единичной дисперсией. Поэтому положительные факторные веса соответствуют тем объектам, которые обладают степенью проявления свойств больше средней. А отрицательные факторные веса соответствуют тем объектам, которые обладают степенью проявления свойств меньше средней. В матричном виде эта система уравнений может быть записана, как

$$\mathbf{X} = \mathbf{S} * \mathbf{F} + \mathbf{E}, \quad (1.2.1)$$

где  $\mathbf{X}$  - матрица признаков (или переменных),  $\mathbf{S}$  - матрица нагрузок,  $\mathbf{F}$  - матрица новых «латентных» переменных,  $\mathbf{E}$  - матрица остатков. Это уравнение описывает



переход от первичных переменных (признаков) к новым переменным (факторам) такое преобразование позволяет:

- Выделить переменные, определяющие исследуемый набор признаков, проанализировать их число и природу.
- Сжать данные – вместо большого объёма переменных система полностью описывается несколькими факторами.

В качестве недостатков этого метода можно перечислить:

1. Нет однозначного подхода к определению числа значимых переменных. Экспериментальные данные, как правило, содержат случайную ошибку, что вызывает появление дополнительных факторов, которые, по сути, бесполезны и описывают погрешность эксперимента. Существует множество способов отделения значимых от незначимых переменных, однако в каждом конкретном случае требуется конкретный подход.
2. Сложность интерпретации переменных – преобразование (1.2.1) можно провести бесконечным множеством способов, при этом выяснить физическую суть каждой новой переменной довольно сложно, а часто и невозможно.

Наиболее распространённые алгоритмы факторного анализа – *метод главных компонент* (principal component analysis, PCA) и *разложение по сингулярным значениям* (singular value decomposition, SVD).

**Центроидный метод** известно основан на предположении о том, что каждый из исходных признаков  $X_i$  ( $i = \overline{1, m}$ ) может быть представлен как функция небольшого числа общих факторов  $F_1, \dots, F_k$  и характерного фактора  $U_j$ . При этом считается, что каждый общий фактор имеет существенное значение для анализа всех исходных признаков, т.е. фактор  $F_j$  - общий для всех  $X_1, \dots, X_m$ . В то же время изменения в характерном факторе  $U_j$  воздействуют на значения только соответствующего признака  $X_i$ . Таким образом, характерный фактор  $U_j$  отражает ту специфику признака  $X_i$ , которая может быть выражена через общие факторы. Основные предположения факторного анализа связаны с допущением о линейности связи исходных признаков с факторами:

$$\begin{aligned}
 X_1 &= a_{11}F_1 + \dots + a_{1k}F_k + d_1U_1 \\
 &\quad \vdots \\
 X_m &= a_{m1}F_1 + \dots + a_{mk}F_k + d_mU_m
 \end{aligned}
 \tag{1.2.2}$$

Общие факторы в модели (1.2.2) предполагаются независимыми стандартизованными показателями, распределёнными по нормальному закону; характерные факторы  $U_1, \dots, U_m$  рассматриваются как некоррелированные стандартизованные показатели, независящие от общих факторов; числа  $a_{ij}$  ( $i = \overline{1, m}; j = \overline{1, k}$ ) - факторные нагрузки, а числа  $d_j, (j = \overline{1, m})$  оценивают степень влияния характерного фактора  $U_j$  на  $X_i$ . Исходные признаки также считаются стандартизованными переменными с нормальным распределением.

В основе каждого метода факторного анализа лежит математическая модель, описывающая соотношения между исходными признаками и обобщёнными факторами. В основе модели в *методе главных компонент* для выражения исходных признаков через факторы лежит предположение о том, что число факторов равно числу исходных признаков ( $k = m$ ), характерные факторы вообще отсутствуют:

$$\begin{aligned}
 X_1 &= a_{11}F_1 + \dots + a_{1m}F_m \\
 &\quad \vdots \\
 X_m &= a_{m1}F_1 + \dots + a_{mm}F_m
 \end{aligned}
 , \tag{1.2.3}$$

где  $a_{ij}$  - факторные нагрузки,  $F_j$  - общие факторы,  $i = \overline{1, m}; j = \overline{1, k}$ . Общие факторы в модели предполагаются независимыми стандартизованными показателями, распределёнными по нормальному закону. Исходные признаки  $X_i$  также считаются стандартизованными переменными с нормальным распределением. Каждая из переменных  $F_j$  называется здесь  $j$ -ой главной компонентой. Очевидно, уравнения (1.2.3) определяют здесь систему преобразования одних параметров в другие. Поскольку число факторов равно числу исходных параметров, задача искомого преобразования решается однозначно, т.е. факторные нагрузки определяются здесь однозначно.

**Метод главных компонент** состоит в построении факторов – главных компонент, каждый из которых представляет линейную комбинацию исходных признаков. Первая главная компонента  $F_1$  определяет такое направление в пространстве исходных признаков, по которому совокупность объектов (точек) имеет наибольший разброс (дисперсию). Вторая главная компонента  $F_2$  строится с таким расчётом, чтобы её направление было ортогонально по отношению к  $F_1$  и она объясняла как можно большую часть остаточной дисперсии, и т.д., вплоть до  $m$ -ой главной компоненты  $F_m$ . Так как выделение главных компонент происходит в убывающем порядке с точки зрения доли, объясняемой ими дисперсии, то признаки, входящие в первую главную компоненту с большими коэффициентами  $a_{1i}$  ( $i = \overline{1, m}$ ), оказывают максимальное влияние на дифференциацию изучаемых объектов.

**Метод экстремальной группировки параметров** также основан на обработке матрицы коэффициентов корреляции между исходными признаками. В основе этого метода лежит гипотеза о том, что совокупность исходных признаков может быть разбита на группы, каждая из которых отражает действие определённого фактора – причины. Поскольку признаки внутри каждой из таких групп должны быть связаны между собой более тесно, чем признаки разных групп, то задача сводится к выявлению «сильно закоррелированных» групп признаков, что позволяет выделить соответствующие факторы.

Формально задача об одновременной группировке параметров и выделении существенных факторов заключается в максимизации как по разбиению параметров на множества  $\{A_1, \dots, A_k\}$ , так и по выбору факторов  $\{F_1, \dots, F_k\}$  одного из критериев.

$$\begin{aligned}
 I_1 &= \sum_{X_i \in A_1} r_{X_i F_1}^2 + \dots + \sum_{X_i \in A_k} r_{X_i F_k}^2, \\
 I_2 &= \sum_{X_i \in A_1} |r_{X_i F_1}| + \dots + \sum_{X_i \in A_k} |r_{X_i F_k}|,
 \end{aligned}
 \tag{1.2.4}$$

где  $r_{X_i F_p}$  коэффициентов корреляции между признаком  $X_i$   $p$ -ой группы и соответствующей ей фактором  $F_p$ , где  $p = 1, \dots, k$ . Таким образом, максимируется сумма квадратов коэффициентов корреляции признаков каждой группы со своим фактором, а во втором случае – сумма модулей этих коэффициентов. Следует отметить связь метода экстремальной группировки параметров с рассмотренными выше методами факторного анализа: метод связанный с максимизацией функционала  $I_1$ , представляет естественное развитие метода главных компонент, а метод связанный с максимизацией функционала  $I_2$  представляет развитие центроидного метода. Так, если группы признаков зафиксированы, то в соответствии с выражением (1.2.4) в пределах каждой группы отыскивается первая главная компонента. Характеризуя особенности этого метода, укажем, что факторы  $F_1, \dots, F_k$ , здесь не общие для всех признаков; каждый из них соответствует «своей» группе признаков. Факторы для этого метода не являются независимыми, ортогональными. Специфика экстремальной группировки параметров состоит в частности в том, что в рамках этого метода каждый признак включается в один из формируемых факторов, в то время как при использовании других методов факторного анализа признаки могут относиться к нескольким факторам сразу или не принадлежать ни к одному из них.

Результаты факторного анализа будут успешными, если удаётся дать содержательную интерпретацию выявленных факторов, исходя из смысла показателей, характеризующих эти факторы. Данная стадия работы требует чёткого представления о содержательном смысле показателей, которые привлечены для анализа и на основе которых выделены факторы.

Большинство методов факторного анализа не статистические в строгом смысле этого слова, так как для них не разработаны способы распространения выборочных результатов на генеральную совокупность. Исходную корреляционную матрицу рассматривают как заданную, а факторы выделяют без учёта ошибки выборки, присущей корреляционной матрице. Исключениями

являются метод максимального правдоподобия (Лоули) и канонический факторный анализ (Рао), для которых разработаны критерии проверки значимости выделенных факторов. При использовании других (основных) методов факторного анализа вопрос значимости факторных нагрузок  $a_{ij}$  обычно решается с помощью эмпирических порогов значимости. Содержательный смысл фактора выявляется на основе признаков, имеющих высокие (значимые) факторные нагрузки.

Одной из проблем факторного анализа является проблема вращения. Любое ортогональное вращение факторов приводит к такой же факторизации с перераспределением нагрузок  $a_{ij}$ , что связано с их неоднозначностью. Необходимость вращения факторов возникает чаще, когда выявленным факторам не удаётся дать достаточно чёткую содержательную интерпретацию. Вращение позволяет сделать матрицу факторных нагрузок более «контрастной» за счёт увеличения нагрузок по одним признакам и уменьшения по другим, что способствует более отчётливому выявлению групп признаков, определяющих тот или иной фактор. Необходимость использования процедур вращения отсутствует в том случае, когда применяют метод экстремальной группировки параметров. Этот метод не связан с ограничением ортогональности факторов, поэтому при его использовании получают факторы, максимально приближенные к «пучкам» взаимосвязанных показателей. В методе экстремальной группировки параметров факторные нагрузки имеют, как правило, весьма высокие значения, так как в этом методе факторные нагрузки признаков, относящихся к одному фактору, зависят от коэффициентов корреляции только между признаками данной группы.

Обратим внимание на оценку полного вклада фактора  $F_p$  в суммарную дисперсию признаков:

$$V_p = a_{1p}^2 + a_{2p}^2 + \dots + a_{mp}^2; p = 1, \dots, k \quad (1.2.5)$$

Полный вклад всех общих факторов будет равен:

$$V = V_1 + V_2 + \dots + V_k \quad (1.2.6)$$

Тогда доля суммарной дисперсии, объясняемой  $k$  факторами, будет равна отношению  $\gamma = \frac{V}{m} 100\%$ ;  $\gamma$  обычно рассматривают как показатель полноты факторизации, т.е. того, насколько хорошо выявленные факторы объясняют вариации исходных признаков.

### 1.2.2 Методы регрессионного анализа

Анализ данных, которые накапливаются с момента начала наблюдения за объектом на ИС и до момента его завершения, приводит к необходимости формировать МО по ним с учётом изменения состояния исследуемого объекта в пределах ИС. Подход, который позволяет выявлять дополнительные характеристики объекта, связанные с поведением объекта в пределах ИС, основан на линейной и нелинейной регрессии и на методе наименьших квадратов (МНК), представленных в рамках теории *регрессионного анализа* [57, 58].

**Регрессионный анализ** – статистический метод, используемый для исследования отношений между двумя величинами.

*Регрессия* в математической статистике – зависимость среднего значения одной величины  $y$  от другой величины (или нескольких величин)  $x$ . В отличие от строгой функциональной зависимости  $y = f(x)$  в регрессионной модели одному и тому же значению величины  $x$  могут соответствовать несколько значений величины  $y$ , иными словами, при фиксированном значении величины  $x$  величина  $y$  имеет некоторое случайное распределение. Регрессионный анализ используется для определения общего вида уравнения регрессии (наиболее часто используется линейная модель), оценки параметров этого уравнения, а также проверки различных статистических гипотез относительно регрессии. Регрессионная модель основана на предположении о том, что величина  $x$  является контролируемой величиной, значения которой задаются во время эксперимента, величина  $y$  является наблюдаемой в ходе эксперимента. Необходимо отметить, что в том случае, если величина  $x$  также является

случайной, то для изучения связи между величинами используются методы корреляционного анализа.

**Метод наименьших квадратов (МНК)** – метод оценки параметров модели на основании экспериментальных данных, содержащих случайные ошибки. В основе метода лежат следующие рассуждения: при замене точного (неизвестного) параметра модели приблизительным значением необходимо минимизировать разницу между экспериментальными данными и теоретическими (вычисленными при помощи предложенной модели). Это позволяет рассчитать параметры модели с помощью МНК с минимальной погрешностью. Мерой разницы в методе наименьших квадратов служит сумма квадратов отклонений действительных (экспериментальных) значений от теоретических значений. Выбираются такие значения параметров модели, при которых сумма квадратов разностей будет наименьшей:  $\sum_i (Y_i - y_i)^2 = \min$ , где  $Y$  - теоретическое значение измеряемой величины,  $y$  - экспериментальное. При этом полученные с помощью МНК параметры модели являются наиболее вероятными. Метод наименьших квадратов, а также его различные модификации (нелинейный МНК, взвешенный МНК и т.д.) широко используются [59, 60].

### 1.2.3 Нейронные сети

В классических системах искусственного интеллекта обработка происходит *последовательно* (sequential), как в традиционном программировании. Даже если порядок выполнения действий строго не определён (например, при сканировании правил и фактов в экспертных системах) операции всё равно выполняются пошагово. В отличие от них, концепция обработки информации в нейронных сетях проистекает из принципа *параллелизма* (parallelism), который является источником их гибкости. Более того, параллелизм может быть массовым (сотни тысяч нейронов), что придаёт нейронным сетям особую форму робастности. Модели искусственного интеллекта – это формальные системы, основанные на использовании языка алгоритмов и представления данных по принципу “сверху

вниз” (top-down), а нейронные сети – это параллельные распределённые процессы, обладающие естественной способностью к обучению и работающие по принципу “снизу вверх” (bottom-up). Поэтому при решении когнитивных задач целесообразно создавать *структурные модели на основе связей* (structured connectionist models) или *гибридные системы* (hybrid system), объединяющие оба подхода. Это обеспечит сочетание свойств адаптивности, робастности и единообразия, присущих нейронным сетям, с представлениями, умозаключениями и универсальностью систем искусственного интеллекта [45]. Для реализации этого подхода были разработаны методы извлечения правил из обученных нейронных сетей [44-48]. Эти результаты позволяют интегрировать нейронные сети с интеллектуальными системами.

*Распознавание образов* формально определяется как процесс, в котором получаемый образ должен быть отнесён к одному из предопределённых классов (категорий). Чтобы нейронная сеть могла решать задачи распознавания образов, её сначала необходимо обучить, подавая последовательность входных образов наряду с категориями, которым эти образы принадлежат. После обучения сети на вход подаётся ранее не виденный образ, который принадлежит тому же набору категорий, что и множество образов, использованных при обучении. Благодаря информации, выделенной из данных обучения, сеть может отнести представленный образ к конкретному классу. Распознавание образов, выполняемое нейронной сетью, является по своей природе статистическим. При этом образы представляются отдельными точками в многомерном пространстве решений. Все пространство решений разделяется на отдельные области, каждая из которых ассоциируется с отдельным классом. Границы этих областей как раз и формируются в процессе обучения. Построение этих границ выполняется статистически на основе дисперсии, присущей данным отдельных классов.

В целом системы распознавания образов, созданные на основе нейронных сетей, можно разделить на два типа.

- Система состоит из двух частей: сети *извлечения признаков* (feature extraction) (без учителя) и сети *классификации* (classification) (с учителем).



Такой метод соответствует традиционному подходу к статистическому распознаванию образов [44, 45]. В концептуальных терминах образ представляется как набор из  $m$  наблюдений, каждое из которых можно рассматривать как точку  $X$  в  $m$ - мерном *пространстве наблюдений (данных)* (observation (data) space). Извлечение признаков описывается с помощью преобразования, которое переводит точку  $X$  в промежуточную точку  $y$  в  $q$ - мерном пространстве признаков, где  $q < m$ . Это преобразование можно рассматривать как операцию снижения размерности (т.е. сжатие данных), упрощающую задачу классификации. Сама классификация описывается как преобразование, которое отображает промежуточную точку  $y$  в один из классов  $r$ - мерного пространства решений (где  $r$  - количество выделяемых классов).

- Система проектируется как единая многослойная сеть прямого распространения, использующая один из алгоритмов обучения с учителем. При этом подходе задача извлечения признаков выполняется вычислительными узлами скрытого слоя сети [44, 45].

При разработке ИРИИС сочетание нейросетевого подхода с другими известными подходами и методами, описанными выше, позволяет, прежде всего, сохранять работоспособность и добиться быстрого действия системы в целом, не теряя при этом в качестве распознавания.

#### **1.2.4 Методы поиска закономерностей (индукция правил)**

Объединение в ИРИИС экспертных систем и нейронных сетей даёт возможность понимать предлагаемую систему как разновидность гибридной интеллектуальной системы. Но в рамках исследований использовался также подход к созданию этой системы как экспертной системы продукционного типа, где *логические алгоритмы классификации (алгоритмы индукции правил)* представляют собой композиции простых легко интерпретируемых правил (закономерностей), выделяющие из обучающей выборки достаточно много объектов какого-то одного класса и практически не выделяющие объекты

остальных классов. Такого типа экспертные системы используют продукционные модели, которые можно считать наиболее распространёнными моделями представления знаний. Системы обработки знаний, использующие продукционные модели, получили название «продукционных систем» [61-63].

Поиск (или синтез) всех закономерностей по заданной выборке данных является NP-полной задачей комбинаторной оптимизации. В общем случае для её решения требуется выполнить полный перебор всех наборов признаков. Поэтому на практике используются эвристические алгоритмы сокращённого перебора. Набор закономерностей в совокупности должен образовывать алгоритм классификации. *Закономерность* или предикат  $\phi(x)$  ( $x$  - объект пространства объектов  $X$ ), удовлетворяющий требованиям *интерпретируемости* и *информативности*. Предикат  $\phi(x)$  считают «хорошо интерпретируемым» или *правилом*, если он описывается простой формулой, понятной экспертам в данной прикладной области. Строгого формального определения *интерпретируемости* не существует. В большинстве практических задач для интерпретируемости необходимо (но не достаточно), чтобы предикат  $\phi(x)$  зависел от небольшого числа  $K$  признаков. Интуитивно предикат  $\phi(x)$  является информативным, если одновременно  $p_y(\phi) \rightarrow \max$  (число положительных объектов класса  $y$  выделенных правилом  $\phi$ ) и  $n_y(\phi) \rightarrow \min$  (число отрицательных объектов класса  $y$  выделенных правилом  $\phi$ ). Кроме того, накладывается ограничение на функциональный вид правил. Чаще всего применяются конъюнкции, реже — шары или гиперплоскости. Другие виды правил будут диктоваться спецификой задачи. В роли  $\phi(x)$  часто выступают конъюнкции небольшого числа элементарных высказываний. Такие *логические* правила имеют вид:

$$\phi(x) = \prod_{j \in \omega} [f_j(x) \leq \geq a_j] \quad (1.2.7)$$

где  $f_j : X \rightarrow D_j, j = 1, \dots, n$  — признаки, с помощью которых описываются объекты пространства  $X$ ,  $\omega$  - подмножество признаков мощности не более  $K$ ,  $a_j$  - пороговое значение признака. Признаки могут быть разнотипными, то есть в общем случае множества  $D_j$  различны. Наиболее распространённый случай - когда все признаки числовые:  $D_j = R$ . При этом числами могут кодироваться признаки, измеренные в различных шкалах, в частности, бинарные, номинальные и порядковые признаки. Для количественных и порядковых признаков используются знаки сравнения  $\leq$  или  $\geq$ ; для бинарных и номинальных — только равенство.

К набору закономерностей, образующих **логический классификатор**, дополнительно предъявляется требование *взаимодополняемости* (различности). Чаще всего логический классификатор представляет собой взвешенную сумму закономерностей:

$$a(x) = \arg \max_{y \in Y} \sum_{t=1}^{T_y} \alpha_{yt} \phi_{yt}(x), \quad (1.2.8)$$

где  $Y$  - множество классов,  $y \in Y$  - фиксированный класс,  $\alpha_{yt}$  - неотрицательные веса. В данной форме могут быть представлены также решающие списки и деревья [61-63].

Требование взаимодополняемости закономерностей означает, что для любого объекта выборки должна найтись закономерность  $\phi_{yt}$ , выделяющая данный объект. В противном случае алгоритм  $a(x)$  не сможет классифицировать объект, то есть произойдёт *отказ от классификации*.

В состав экспертных систем продукционного типа входят: база правил (знаний), рабочая память и интерпретатор правил (решатель), реализующий определённый механизм логического вывода. Любое продукционное правило, содержащееся в базе знаний, состоит из двух частей: *антецедента* и *консеквента*. Антецедент представляет собой посылку правила (условную часть) и состоит из элементарных предложений, соединённых логическими связками «и», «или».

Консеквент (заключение) включает одно или несколько предложений, которые выражают либо некоторый факт, либо указание на определённое действие, подлежащее исполнению. Продукционные правила принято записывать в виде *антецедент-консеквент*. Любое правило состоит из одной или нескольких пар «атрибут-значение». В рабочей памяти систем, основанных на продукционных моделях, хранятся пары атрибут-значение, истинность которых установлена в процессе решения конкретной задачи к некоторому текущему моменту времени. Содержимое рабочей памяти изменяется в процессе решения задачи. Это происходит по мере срабатывания правил. Правило срабатывает, если при сопоставлении фактов, содержащихся в рабочей памяти, с антецедентом анализируемого правила имеет место совпадение, при этом заключение сработавшего правила заносится в рабочую память. Поэтому в процессе логического вывода объём фактов в рабочей памяти, как правило, увеличивается (уменьшаться он может в том случае, если действие какого-нибудь правила состоит в удалении фактов из рабочей памяти). В процессе логического вывода каждое правило из базы правил может сработать только один раз [62-65].

Существуют два типа продукционных систем – с «**прямыми**» и «**обратными**» выводами. Прямые выводы реализуют стратегию «от фактов к заключениям». При обратных выводах выдвигаются гипотезы вероятностных заключений, которые могут быть подтверждены или опровергнуты на основании фактов, поступающих в рабочую память. Существуют также системы с двунаправленными выводами [61-63].

Создание продукционной системы на основе нечёткой логики позволяет более адекватно представлять практические знания экспертов и строить методы и схемы вывода заключений на основе нечётких условий в базе правил нечётких продукций. В настоящее время наибольшее применение в системах нечёткого вывода получили алгоритмы: Мамдами; Цукамото, Ларсена и Сугено [66-68]. При решении практических задач нечёткого моделирования могут быть использованы одновременно несколько алгоритмов для получения наилучших результатов.

Основные достоинства систем, основанных на продукционных моделях, связаны с простотой представления знаний и организации логического вывода. К недостаткам таких систем можно отнести следующее:

- отличие от структур знаний, свойственных человеку;
- неясность взаимных отношений правил;
- сложность оценки целостного образа знаний;
- низкая эффективность обработки знаний.

При разработке небольших систем (десятки правил) проявляются в основном положительные стороны продукционных моделей знаний, однако при увеличении объёма знаний более заметными становятся слабые стороны [64, 65].

### **1.3 ВЫВОДЫ**

Основной задачей для исследований данной работы стала разработка таких принципов организации и построения ИРИИС на основе РВОИС большой размерности сегментарного типа, которая обеспечивает при достаточно хорошем качестве распознавания динамических образов высокую скорость обработки данных для принятия управленческих решений в реальном времени.

При реализации эта система должна обладать следующими качествами и свойствами:

- 1) система должна работать в режиме реального времени;
- 2) система должна обслуживать одновременно достаточно большое количество ИС РВОИС при мониторинге;
- 3) система должна обладать возможностью адаптации к конкретным условиям и пользователям (возможность модификации базы знаний);
- 4) система должна соответствовать требованиям, предъявляемым к современным информационно-измерительным системам;
- 5) система должна обладать гибкостью и высокой скоростью принятия решений при распознавании динамических образов.

Ограничения для ИРИИС:

- 1) количество объектов одновременного мониторинга;

## 2) ограниченное количество типов идентифицируемых объектов

Анализ методов показывает целесообразность комплексного подхода к решению проблемы распознавания динамических образов. Необходимо использовать различные методы классификации образов и подходы построения интеллектуальных систем, комбинирование нейронных сетей в сочетании с другими соответствующими средствами, что обеспечит робастность и гибкость ИРИИС.

Таким образом, ИРИИС должна представлять собой разновидность гибридной интеллектуальной системы с такой организацией, которая предусматривает наличие модуля для выбора разных подходов моделирования системы при распознавании динамических образов в зависимости от полученного типа модели образа.

## 2. РАСПРЕДЕЛЁННАЯ ВОЛОКОННО-ОПТИЧЕСКАЯ ИЗМЕРИТЕЛЬНАЯ СЕТЬ СЕГМЕНТАРНОГО ТИПА

Совершенствование ИИС для распознавания динамических образов может осуществляться, с одной стороны, за счёт создания новых датчиков внешних физических воздействий измерительных сетей и оптимизации построения самой сети, обеспечивая этим улучшение метрологических характеристик системы [1] [69-73]. С другой стороны, важным направлением развития такого рода систем является совершенствование систем обработки данных, разработка алгоритмических комплексов для моделирования и распознавания динамических образов по выделенным признакам объекта в условиях помех и шума, а также при влиянии окружающей среды на сигналы измерительной системы [74-76]. В данной работе в качестве измерительной сети предлагается использовать РВОИС сегментарного типа, которая может состоять из большого количества различных датчиков физических величин, объединенных в единую измерительную сеть [51]. Преимущества такой сети как РВОИС сегментарного типа перед сетями, построенными по иным принципам, прежде всего, определены устойчивостью волоконно-оптических датчиков к электромагнитным помехам, агрессивным воздействиям окружающей среды и высокой их чувствительностью, обусловленной интерференционным механизмом отклика [1, 78]. Такого рода сети могут быть замаскированы в грунте при охране периметра или встроены в конструкцию протяжённого объекта для осуществления контроля состояния в режиме реального времени [79, 80]. Сегментарный тип РВОИС позволяет разделять информационные потоки данных по сегментам и тем самым повышает функциональность и скорость работы ИРИИС. Физическая независимость сегментов друг от друга предусматривает возможность сохранять работоспособность и восстанавливать ИРИИС при частичном её повреждении. Каждый сегмент РВОИС представляет собой практически аналог измерительной сети, который можно использовать независимо от других. Принципы организации сегмента РВОИС зависят от выбора вида ВОД и тех функций, которые должна

выполнять ИРИИС. Применение протяжённых интегрирующих датчиков, выходной сигнал которых пропорционален интегральному воздействию исследуемого поля вдоль линии укладки, позволяет повысить помехозащищённость системы. Такие датчики, по сути, являются измерительными линиями, которые могут быть легко объединены в распределённую измерительную сеть [51, 78-80].

Основные принципы организации и построения ИРИИС на основе РВОИС большой размерности сегментарного типа использовались при разработке макета измерительной сети волоконно-оптической системы охраны периметра [51, 78-81].

## **2.1 ВИДЫ УКЛАДКИ ВОЛОКОННО-ОПТИЧЕСКИХ ЛИНИЙ В РАСПРЕДЕЛЁННОЙ ВОЛОКОННО-ОПТИЧЕСКОЙ ИЗМЕРИТЕЛЬНОЙ СЕТИ СЕГМЕНТАРНОГО ТИПА**

Разнообразие физических полей, порождаемых ими распределений их характеристик, делает необходимым исследовать различные способы укладки волоконно-оптических линий в РВОИС [1, 82]. В качестве базовой схемы расположения точек измерения и реконструкции сигналов РВОИС для удобства принята квадратная карта, задаваемая количеством точек  $n$ , приходящихся на одну ее сторону (общее количество контрольных точек измерительной системы  $m = n^2$ ).

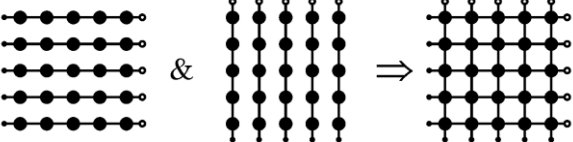
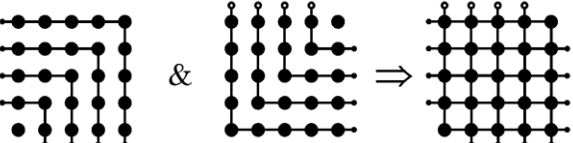
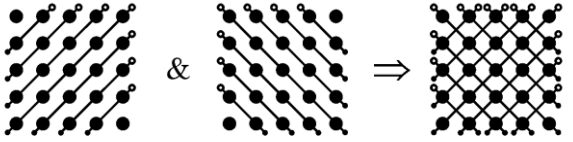
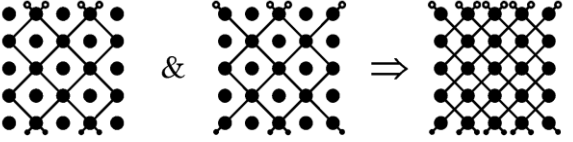
В таблице 2.1 на примере карты размерностью  $5 \times 5$ , представлены базовые схемы исследуемых в работе измерительных сетей, их условные обозначения, а также формулы для вычисления количества входящих в них измерительных линий. Данные структуры определены как базовые, поскольку путем их объединения могут быть получены более сложные схемы, также представляющие интерес в рамках проводимого исследования. Для обозначения таких схем будем использовать символ конъюнкции, например: "Клетка" & "Диагональ".

При проведении настоящего исследования целесообразно абстрагироваться от широкого спектра физико-технических вопросов построения измерительных



сетей и сосредоточиться исключительно на математической составляющей задачи реконструкции в одной из ее наиболее распространенных постановок [83, 84].

Таблица 2.1. Принятые в рассмотрение базовые структуры измерительных сетей

Условное обозначение	Пример последовательного формирования структуры измерительной сети для случая $n = 5$	Количество измерительных линий
"Клетка"		$h = 2n$
"Угол"		$h = 2(n - 1)$
"Диагональ"		$h = 2(2n - 3)$
"Ромб"		$h = 2(n - 1)$

Пусть  $F(x_i, y_i) = z_i$  – распределенное на плоскости физическое поле, представленное амплитудой  $z_i$  в дискретных точках пространства с координатами  $x_i$  и  $y_i$ ,  $i = \overline{1, m}$ , где  $m$  – число контрольных точек. Тогда интегральные показания  $S_j$ , поступающие от измерительных линий, могут быть записаны в виде суммы значений интенсивности поля в точках:

$$S_j = \sum_{k=1}^{a_j} z_{B_{k,j}}, \quad (2.1.1)$$

где  $j = \overline{1, h}$  и  $h$  – порядковый номер и общее количество измерительных линий соответственно,  $B_{k,j}$  – порядковый номер  $k$ -ой контрольной точки, входящей в  $j$ -ю измерительную линию,  $a_j$  – общее количество контрольных точек в  $j$ -й

измерительной линии. Параметр  $h$ , элементы  $a_j$  вектора  $\mathbf{a}$  и элементы  $B_{k,j}$  матрицы  $\mathbf{B}$  определяются схемой укладки измерительных линий.

Решение задачи реконструкции распределенного физического поля  $z_1, \dots, z_m$  состоит в восстановлении его исходного распределения  $F(\cdot)$  в контрольных точках по интегральным данным  $s_1, \dots, s_h$ , поступающим с измерительных линий. Совокупность выражений (2.1.1) образует систему уравнений, которую удобно представить в матричном виде

$$\mathbf{Az} = \mathbf{s}, \quad (2.1.2)$$

где  $\mathbf{A}$  – матрица измерительной системы,  $\mathbf{z} = (z_1, \dots, z_m)$  – вектор ее решения,  $\mathbf{s} = (s_1, \dots, s_h)$  – вектор свободных членов. Решение этой системы относительно линейно входящих в нее неизвестных  $z_1, \dots, z_m$  приведет к решению задачи реконструкции.

На рисунке 2.1 представлен пример измерительной сети размерностью  $2 \times 2$ .

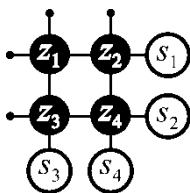


Рис. 2.1. Пример измерительной сети и ее индексации

Матрица  $\mathbf{A}$ , векторы  $\mathbf{z}$  и  $\mathbf{s}$  для этого примера измерительной сети

$$\mathbf{A} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}; \quad \mathbf{z} = \begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{pmatrix}; \quad \mathbf{s} = \begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{pmatrix}. \quad (2.1.3)$$

В рассмотренном примере число неизвестных  $m$  равно числу уравнений  $h$ . Выполнение этого условия является необходимым для возможности однозначного аналитического решения такой системы, что в целом делает решение задачи реконструкции тривиальным. Вместе с тем, с практической точки зрения, построение таких измерительных систем ( $m = h$ ) нецелесообразно.

Положительный эффект от интегральной реализации измерительной системы достигается при выполнении условия  $m > h$ . Это не отражается на способе формализации задачи реконструкции (2.1.2). Однако такая система уравнений уже не имеет однозначного аналитического решения. Это положение составляет основную сложность решения задачи реконструкции.

### 2.1.1 Ранг матрицы измерительной системы, как первичный критерий качества укладки интегральных линий

Каждая из рассматриваемых в работе схем укладки измерительных линий образует свою матрицу  $\mathbf{A}$ . Анализ этих матриц показал, что в большинстве случаев их ранг, определяющий количество линейно независимых уравнений системы, оказывается меньше числа измерительных линий  $h$ . С практической точки зрения это означает, что в структуре измерительной сети присутствуют по существу дублирующие линии, не несущие никакой дополнительной информации. Причем с учетом разреженного характера матрицы  $\mathbf{A}$  вероятность обнаружения таких "дублирующих" линий чрезвычайно высока. Результаты соответствующего анализа, выполненного для случая  $n = 10$  ( $m = 100$ ) и различных структур измерительных сетей, представлены в таблице 2.2.

Таблица 2.2. Количество интегральных линий и ранг матриц измерительных систем

Измерительная сеть	Измерительных линий, $h$	rank( $\mathbf{A}$ )
"Клетка"	20	19
"Клетка" & "Диагональ"	54	51
"Клетка" & "Ромб"	38	36
"Угол" & "Диагональ"	52	48

Во всех представленных случаях выявленную проблему можно решить путем внесения изменений в топологию сети. В рамках проводимого исследования для этих целей использован алгоритм последовательного обнуления ненулевых элементов матрицы  $\mathbf{A}$  и проверки ее ранга. Если такое обнуление приводит к повышению ранга, то оно сохраняется ( $\mathbf{A}^*$ ), иначе - игнорируется.

Выполнение алгоритма прекращается по выполнению условия  $\text{rank}(\mathbf{A}^*) = h$ . Например, результат его применения для матрицы (2.1.3):

$$\mathbf{A}^* = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}.$$

Оптимизированную структуру измерительной сети легко восстановить по виду полученной в итоге матрицы. Обратим внимание, что во всех последующих экспериментах, за исключением специально оговоренных случаев, будут использоваться только оптимизированные структуры измерительных сетей.

### 2.1.2 Основные механизмы аналитического решения задачи реконструкции

Задача реконструкции распределенного физического поля по неполным интегральным данным в общем случае сводится к решению системы алгебраических уравнений неполного ранга. При работе с такой системой найдется бесконечное число решений, удовлетворяющих тождеству (2.1.2). Для преодоления проблемы неоднозначности, исходную постановку задачи можно дополнить априорными знаниями, накладывающими ограничения на вид допустимого решения. Такой прием позволяет либо сократить количество неизвестных исходной системы уравнений, либо увеличить число входящих в нее линейно независимых уравнений (неравенств).

### 2.1.3 Механизм сокращения числа неизвестных измерительной системы: реконструкция точечных распределений

Пример точечного распределения представлен на рисунке 2.2 ( $n = 20$ ). Для определения этого распределения будем использовать параметр  $\nu$ , характеризующий число точек, в которых значение амплитуды распределенного физического поля отлично от нуля (для представленного примера  $\nu = 5$ ).

Введем в решение допущение о постоянстве знака распределения. Это допущение исключает возможность компенсации амплитуд различных знаков,

что позволяет рассматривать все контрольные точки  $z_j$ , охваченные измерительными линиями с нулевыми показаниями  $S_i$ , как точки, имеющие заведомо нулевые значения. Очевидно, что такие точки могут быть исключены из числа неизвестных системы. Если площадь ненулевой части распределения достаточно мала, то данный прием позволяет привести число неизвестных системы к ее рангу. Эксперименты показывают, что успех этой операции зависит не только от числа ненулевых точек  $\nu$ , но и от их взаимного расположения.

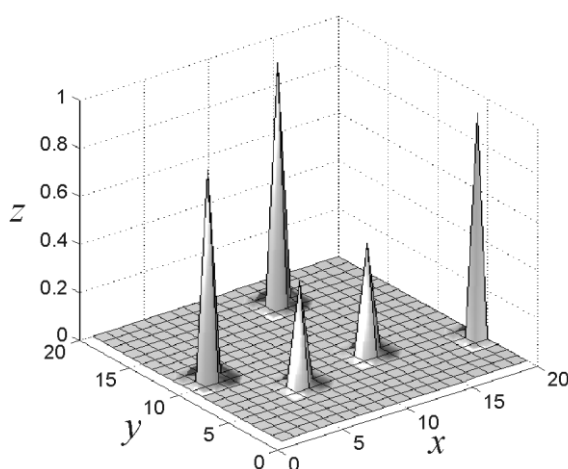


Рис. 2.2. Пример точечного распределения

На рисунке 2.3 (а) изображены результаты аналитической реконструкции точечных распределений, полученные на базе измерительных сетей различной структуры (для всех экспериментов принято  $n = 10$ ). По оси абсцисс отложено значение параметра  $\nu$ , а по оси ординат - вероятность однозначной реконструкции такого распределения. Каждая точка представленных кривых получена путем обобщения результатов реконструкции 1000 распределений с заданным  $\nu$  и произвольным положением ненулевых точек. Пунктирной линией изображена кривая, соответствующая измерительной сети с неоптимизированной структурой.

В рассмотренном эксперименте допущение о постоянстве знака распределения используется лишь частично, на этапе редуцирования исходной матрицы  $\mathbf{A}$ . При этом само решение системы уравнений не накладывает никаких

знаковых ограничений. Поэтому реализованный подход фактически соответствует менее сильному ограничению, которое лишь исключает вероятность компенсации положительных значений распределения отрицательными значениями, при допущении одновременного существования таковых.

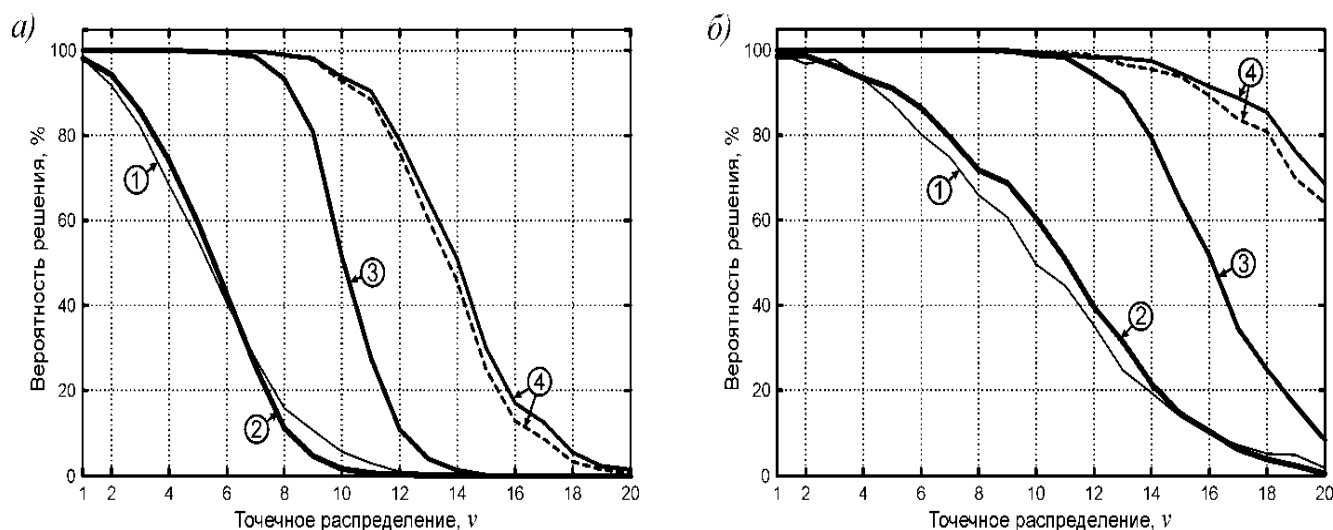


Рис. 2.3. Результаты реконструкции точечных распределений

( $n = 10; m = 100$ ):

- (1) – "Угол" & "Диагональ",  $h = 52$ ; (2) – "Угол" & "Ромб",  $h = 36$ ;  
 (3) – "Клетка" & "Ромб",  $h = 38$ ; (4) – "Клетка" & "Диагональ",  $h = 54$

На рисунке 2.3 (б) представлены результаты экспериментов, где допущение о постоянстве знака распределения используется в полном объеме. Реализация этого решения требует привлечения более затратных итерационных методов численной оптимизации [85].

По результатам проведенных экспериментов можно сделать вывод, что интегральная реализация измерительной системы обеспечивает разрешающую способность  $m$  при числе измерительных линий  $h < m$ , что недостижимо в системах непосредственного измерения в разреженных точках при аналогичных ограничениях на число измерительных каналов. Однако платой за это преимущество выступает ограниченное количество надежно регистрируемых

точек  $v$ , число которых оказывается существенно меньше числа измерительных линий  $h$ .

### 2.1.4 Механизм увеличения числа уравнений измерительной системы: реконструкция гладких распределений

В работе выполнено математическое исследование задачи реконструкции распределенных физических полей в измерительных системах интегрального типа. Продемонстрирована возможность и основные пути аналитического решения и анализа этой задачи. Выявлена необходимость ранговой проверки матриц измерительных систем, а так же показана линейность решения задач реконструкции гладких дифференцируемых распределений.

Примеры исследуемых в работе гладких распределений представлены на рисунке 2.4. В качестве базовых элементов при их конструировании приняты степенной и тригонометрический ряды со случайным выбором коэффициентов. Использование рядов позволяет легко регулировать сложность таких распределений путем изменения числа их членов  $p$ .

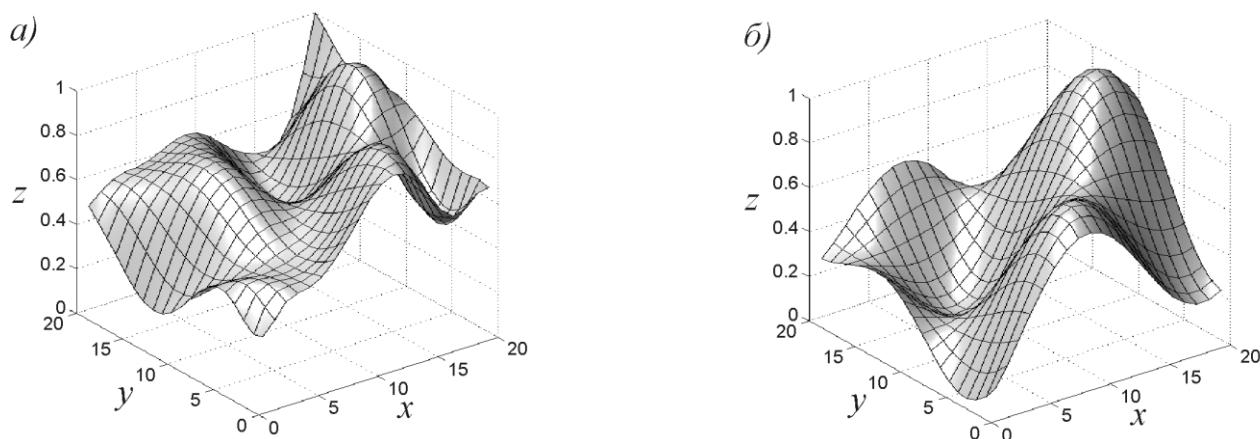


Рис. 2.4. Примеры гладких распределений, построенных на базе:

а) степенного ( $p = 5$ ) и б) тригонометрического ( $p = 2$ ) рядов

Механизм увеличения числа уравнений измерительной системы также базируется на исходных допущениях о типе реконструируемого распределения. В данном случае в качестве такого допущения выступает заведомо гладкий характер

поверхности распределения. Понятие гладкости непосредственно связано с понятием дифференцируемости, а оно, в свою очередь, может быть выражено вполне конкретными количественными соотношениями.

Рассмотрим распределение  $F(x, y)$  в непрерывном пространстве координат  $x$  и  $y$ . Пусть в области определения этих координат для некоторого  $y$  и любого  $x$  выполняется условие  $dF(x, y)/dx = \text{const}$ , т.е. амплитуда распределения  $F(x, y)$  в сечении  $y$  монотонно возрастает или убывает. Тогда амплитуда распределения  $F(x_1, y)$  в любой точке  $x_1$  может быть выражена через амплитуды распределения  $F(x_2, y)$  и  $F(x_3, y)$  в паре любых других точек ( $x_2 \neq x_3$ ) сечения  $y$ :

$$F(x_1, y) = F(x_2, y) + (x_1 - x_2) \frac{dF(x, y)}{dx} = F(x_2, y) + (x_1 - x_2) \frac{F(x_2, y) - F(x_3, y)}{x_2 - x_3}.$$

Распространяя это геометрическое свойство распределения на случай дискретного пространства координат измерительной системы:  $z_1 = F(x_1, y)$ ,  $z_2 = F(x_2, y)$ ,  $z_3 = F(x_3, y)$ , где  $x_1 < x_2 < x_3$ ,  $x_1 - x_2 = x_2 - x_3$ , получим  $z_1 - 2z_2 + z_3 = 0$ . Аналогичным образом можно записать выражения, учитывающие дифференциальные характеристики более высоких порядков. Таким образом, априорные знания о гладкости (дифференцируемости) реконструируемого распределения могут быть введены в решение явно, в виде дополнительных уравнений измерительной системы, которые устанавливают количественную связь между отдельными точками реконструируемого распределения и одновременно приводят ее матрицу  $\mathbf{A}$  к аналитически разрешимому квадратному виду.

Рассматриваемый способ формализации априорных предположений о гладкости реконструируемого распределения обладает широкой свободой, как в отношении учитываемого порядка дифференцирования, так и в отношении выбора групп контрольных точек, чьи показания количественно связываются. Вместе с тем, численные эксперименты показали, что результат реконструкции существенно зависит от этого выбора, и для достижения его наилучших показателей требуется учет индивидуальных геометрических особенностей реконструируемого класса гладких распределений. Для количественного



выявления этих особенностей целесообразно обратиться к статистическим методам.

Решение матричного уравнения (2.1.2) можно записать в виде  $\mathbf{z} = \mathbf{A}^{-1}\mathbf{s}$ , где  $\mathbf{A}^{-1}$  – матрица, обратная к матрице  $\mathbf{A}$ . Вид этих матриц в данном случае полностью определяется схемой укладки измерительных линий и способом формализации исходных допущений. Эти параметры остаются неизменными в процессе функционирования измерительной системы. Поэтому матрица  $\mathbf{A}^{-1}$  может быть определена единожды, на этапе синтеза системы, для чего необходимо предварительно подготовить выборку, состоящую из примеров распределений того же класса, что планируется реконструировать в процессе рабочего функционирования измерительной системы. Искомые коэффициенты удобно определить с помощью метода наименьших квадратов, из условия

$$\sum_{i=1}^N \sum_{j=1}^m (z_{i,j} - z_{i,j}^*)^2 \rightarrow \min, \quad (2.1.4)$$

где  $N$  – количество примеров в выборке,  $m$  – число контрольных точек измерительной системы,  $z_{i,j}$  и  $z_{i,j}^*$  – соответственно оригинальная и реконструированная амплитуда распределения в  $j$ -ой контрольной точке  $i$ -го примера выборки.

Погрешности, возникающие при реконструкции исследуемых в работе гладких распределений (пример на рисунке 2.4), сведены в таблицу 2.3. Представленные показатели выражены в процентах относительно максимальной амплитуды распределений, которая во всех экспериментах установлена равной единице. Они получены путем усреднения погрешностей реконструкции 1000 различных распределений, не участвующих в процессе параметризации матриц  $\mathbf{A}^{-1}$ , но относящихся к одному классу. Исходная параметризация матриц  $\mathbf{A}^{-1}$  для каждого случая выполнена на аналогичных по объему выборках ( $N = 1000$ ).

Кроме результатов систем интегрального типа, в рассматриваемой таблице представлены результаты реконструкции, полученные для систем

непосредственного измерения в разреженных точках физического поля (равномерно распределены по поверхности измерения):

$$z_i^* = \sum_{j=1}^{h_d} w_{i,j} \cdot z_{e_j}, \quad (2.1.5)$$

где  $i$  – порядковый номер реконструируемой контрольной точки  $z_i^*$ ,  $\mathbf{w}$  – матрица постоянных коэффициентов,  $h_d$  – общее количество выборочных точек непосредственного измерения (число измерительных каналов),  $c_j$  – порядковый номер  $j$ -й контрольной точки, подлежащей непосредственному измерению. Так же как и ранее, матрица коэффициентов  $\mathbf{w}$  для каждого случая получена путем аналитической минимизации функционала (2.1.4);  $N = 1000$ .

Таблица 2.3. Усредненные погрешности реконструкции гладких распределений в измерительных системах различного типа ( $n = 25$ ), %

Тип измерительной системы ( $n = 25$ ; $m = 625$ )	Измерительных каналов, ( $h$ или $h_d$ )	Тип распределения				
		Степенной ряд		Тригонометрический ряд		
		$p = 5$	$p = 10$	$p = 5$	$p = 10$	$p = 15$
Сеть: "Ромб"	48	6.87	8.01	6.11	6.26	6.42
Сеть: "Клетка"	50	4.72	6.94	9.22	9.73	10.2
Сеть: "Диагональ"	94	2.99	5.53	2.95	2.97	3.03
Сеть: "Клетка" & "Ромб"	98	0.32	4.05	3.42	4.25	4.49
Сеть: "Угол" & "Диагональ"	142	0.89	4.14	2.31	2.33	2.42
Сеть: "Угол" & "Диагональ"*	142	2.91	4.79	2.65	2.55	2.57
Сеть: "Клетка" & "Диагональ"	144	$6 \cdot 10^{-11}$	2.85	1.81	2.09	2.29
Прямое измерение в точках	25	0.19	4.49	7.61	10.1	11.4
Прямое измерение в точках	49	$8 \cdot 10^{-13}$	0.77	1.73	7.63	8.91
Прямое измерение в точках	144	$2 \cdot 10^{-13}$	$3 \cdot 10^{-12}$	0.006	0.48	4.54

\* Неоптимизированная структура измерительной сети ( $h = 142$ ;  $\text{rank}(\mathbf{A}) = 138$ )

Анализ представленных в таблице 2.3. результатов показывает, что в большинстве рассмотренных случаев качество реконструкции гладких распределений по ограниченному числу непосредственных измерений в точках

превосходит качество их реконструкции по неполным интегральным данным. Допущение о гладкости реконструируемого распределения позволяет отказаться от измерения поля во всех его точках. При этом точки, чье прямое измерение не выполнено, могут быть восстановлены методами интерполяции или экстраполяции. Как показывает сравнительный эксперимент, в этом случае такие методы оказались точнее и надежнее, чем выделение частных характеристик распределения из общей интегральной картины. Исключение составляет лишь последний столбец таблицы, где представлены результаты реконструкции распределения с наивысшей пространственной частотой (наибольшее число членов тригонометрического ряда). Здесь разреженность точек в системах непосредственного измерения приводит к потере значимой информации и, как следствие, снижению качества интерполяции (экстраполяции) не измеряемых точек. Рассмотренный эксперимент свидетельствует, как о более высокой разрешающей способности измерительных систем интегрального типа, так и о неэффективности их применения в тех случаях, когда эта способность оказывается невостребованной.

В работе выполнено математическое исследование задачи реконструкции распределенных физических полей в измерительных системах интегрального типа. Продемонстрирована возможность и основные пути аналитического решения и анализа этой задачи. Выявлена необходимость ранговой проверки матриц измерительных систем, а так же показана линейность решения задач реконструкции гладких дифференцируемых распределений.

Сравнение систем непосредственного измерения в разреженных точках распределенного физического поля и систем интегрального типа, выполненное на примерах регистрации точечных и гладких распределений, позволило выявить область целесообразного применения последних. В первую очередь она просматривается там, где возникает потребность в высокой разрешающей способности измерительных систем. В противном случае целесообразность применения систем интегрального типа может привести к усложнению задачи,

снижению качества и надежности ее решения, а поэтому требует отдельного обоснования.

## **2.2 ФИЗИЧЕСКИЕ ПРИНЦИПЫ ФУНКЦИОНИРОВАНИЯ РАСПРЕДЕЛЁННОЙ ВОЛОКОННО-ОПТИЧЕСКОЙ СЕТИ СЕГМЕНТАРНОГО ТИПА**

### **2.2.1 Общие принципы построения интегрирующих волоконно-оптических измерительных линий**

Интегрирующая волоконно-оптическая измерительная линия (ВОИЛ) представляет собой протяженный датчик способный регистрировать внешнее воздействие по всей его длине. Как показано в работе [34], выходной сигнал интегрирующей волоконно-оптической измерительной линии пропорционален контурному интегралу от исследуемого физического поля  $f(x,y)$  при контуре интегрирования, проходящем вдоль ВОИЛ:

$$h(\rho, \varphi) = k \int_{L(\rho, \varphi)} f(x, y) dl = k \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \cdot \delta[\rho - x \cos(\varphi) - y \sin(\varphi)] dx dy, \quad (2.2.1)$$

где  $\rho$ ,  $\varphi$  - полярные координаты, задающие положение контура  $L$ :  $\rho$  - расстояние от начала координат до прямой  $L$ ,  $\varphi$  - угол между  $\rho$  и осью абсцисс (рис. 2.5),  $k$  - коэффициент пропорциональности между величиной сигнала на выходе измерительной линии и величиной внешнего воздействия со стороны физического поля.

Выражение (2.2.1) в точности совпадает с хорошо изученным в теории томографии выражением, определяющим преобразование Радона -  $\hat{R}$  [53, 54], которое ставит в соответствие двумерной функции  $f(x,y)$  (оригиналу) ее интегральный образ:  $h(\rho, \varphi) = \hat{R}[f(x, y)]$ . По дискретному интегральному образу, формируемому распределенной измерительной сетью, состоящей из набора интегрирующих ВОИЛ, возможна реконструкция исследуемой функции  $f(x,y)$ .

Тип измерительного преобразователя интегрирующей волоконно-оптической измерительной линии зависит от того, какой именно параметр оптического излучения (амплитуду, фазу, частоту, состояние поляризации) необходимо модулировать. Конструкция чувствительного элемента определяется характером регистрируемой физической величины и может быть различной [88-91].

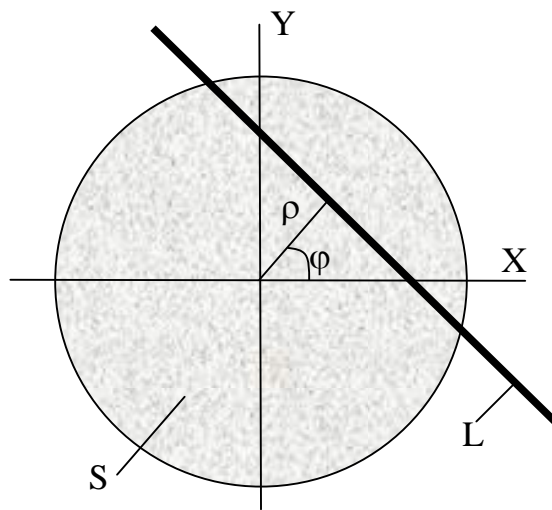


Рис. 2.5. Схема укладки волоконно-оптической измерительной линии. S – область распределения физического поля

Входной волоконный световод (ВС), направляющий оптическое излучение от источника в измерительный преобразователь, должен быть нечувствителен к внешним воздействиям. В противном случае отношение сигнал/шум ВОИЛ становится неприемлемым для обеспечения эффективной работы информационно-измерительной системы. Способы решения этой проблемы определяются измеряемой физической величиной и типом измерительного преобразователя. В большинстве случаев они связаны с необходимостью применения ВС другого типа, нежели в измерительном преобразователе, что порождает новую проблему – проблему согласования входного ВС и ВС преобразователя.

Выходной ВС также должен быть защищен от нежелательных внешних воздействий, что порождает те же проблемы, что и для входного ВС.

Обычно РВОИС используют именно для задач восстановления картины физических полей и в таких случаях практический интерес представляют фазовые измерительные преобразователи с высокой чувствительностью измерительной линии к измеряемой физической величине, а амплитудные и поляризационные преобразователи не обеспечивают необходимой точности измерений, частотные же сложны в реализации [1]. При использовании фазовых измерительных преобразователей в состав ВОИЛ необходимо включить устройство фазовой демодуляции для преобразования фазовой модуляции в амплитудную модуляцию, которая затем и регистрируется фотоприемником. Измерительный преобразователь интегрирующей ВОИЛ может содержать дополнительные преобразователи – чувствительные элементы, усиливающие чувствительность ВС к измеряемой физической величине.

Так как в данной работе решалась проблема не реконструкции исследуемой функции  $f(x,y)$ , а идентификации движущегося объекта по воздействию на РВОИС и определения его основных характеристик, случайным образом изменяющихся с течением времени, поэтому возникла необходимость разработать чувствительный элемент ВОИЛ, позволяющий улавливать характер изменения регистрируемых сигналов РВОИС и по нему формировать модель динамического образа.

Для реализации основных идей, а также, чтобы работа имела практический интерес, была разработана ИРИИС для распознавания динамических образов при движении объектов в зоне мониторинга РВОИС. Для регистрации движущихся объектов использованы чувствительные элементы ВОИЛ, позволяющие достигнуть поставленных целей при обработке полученных данных РВОИС.

### **2.2.2 Волоконно-оптические датчики на основе изгиба оптоволокна**

Волоконно-оптические датчики на основе изгиба ВС относятся к амплитудным волоконно-оптическим датчикам, в которых в результате внешнего физического воздействия наблюдается непосредственная модуляция интенсив-

ности распространяющихся по ВС оптических сигналов. Они являются наиболее простыми и удобными в эксплуатации конструкциями ВОД. Действительно, большинство амплитудных ВОД не требует использования когерентных источников света, а их измерительные схемы не предъявляют специальных требований ни к источникам, ни к приемникам излучения, а также не нуждаются в дополнительной обработке выходных сигналов датчиков. К настоящему моменту разработаны разнообразные конструкции амплитудных ВОД физических величин, которые условно можно разделить на два основных класса [1].

К первому классу датчиков относятся амплитудные ВОД, в которых волоконные световоды выполняют пассивную функцию, связанную только с подводом и отводом излучения от чувствительного элемента (ЧЭ). Такого рода конструкции имеют высокую чувствительность и достаточно просты, однако обладают рядом недостатков, которые не позволяют использовать их в распределенных измерительных системах. Эти недостатки кроются в необходимости разрыва непрерывной волоконной линии для обеспечения ввода излучения в чувствительный элемент датчика, что приводит к значительным потерям световой мощности на элементах межсоединений, кроме того, использование разнородных оптических компонентов обуславливает низкую механическую стабильность характеристик измерительных устройств.

ВОД второго класса базируются на использовании непрерывных волоконных линий, когда ВС является главной составляющей чувствительного элемента датчика. В таких датчиках для измерения величины внешнего воздействия используются явления, приводящие к эффекту модуляции интенсивности распространяющегося излучения по ВС. К этим явлениям относятся: зависимость эффективности процесса Рэлеевского рассеяния излучения в ВС от внешних условий; нарушение явления полного внутреннего отражения и направленная связь мод, вызываемые внешним воздействием. Использование в таких датчиках «непрерывных световодных линий» открыло возможность объединения их в протяженные распределенные измерительные линии, в которых мультиплексирование/демультиплексирование сигналов производится с

использованием хорошо известного в волоконно-оптических системах связи метода импульсно-временной рефлектометрии.

Потери мощности при прохождении оптического излучения в изогнутом ВС исследуются подробно как теоретически, так и экспериментально на протяжении многих лет. Большой интерес представляет практическое использование изменения потерь мощности излучения, т.е. амплитудная модуляция путем изгиба оптоволокна в датчиках различного назначения [1, 92-96].

Для представленной в данной работе ВОИС были разработаны ЧЭ особой конструкции, преобразующие регистрируемое воздействие (давление на чувствительный элемент) в макроизгиб ВС, приводящий к потерям мощности направляемого излучения.

Механизм образования изгибных потерь наиболее просто поясняется построением хода лучей в волокне согласно представлениям геометрической оптики (рис. 2.6). По сердцевине ВС могут распространяться только лучи, угол падения которых на границу сердцевина-оболочка  $\theta \geq \theta_c = \arcsin(n_2/n_1)$ , где  $\theta \geq \theta_c$  - угол полного внутреннего отражения на границе сердцевина оболочка;  $(n_2/n_1)$  - коэффициенты преломления соответственно сердцевины 1 и оболочки 2.

Если  $\theta < \theta_c$ , лучи выходят в оболочку 2 волокна, то происходит перекачка энергии из направляемых мод в моды оболочки и вытекающие моды. В свою очередь, моды оболочки при распространении вдоль ВС сами могут постепенно преобразовываться в вытекающие моды.

Более детальное представление об особенностях распространения излучения внутри изогнутых ВС получают использованием электродинамического подхода [101-104]. Одной из основных особенностей изогнутых ВС является появление избыточных потерь [94-100].

Этот эффект вызван излучением мод на участке изгиба и их конверсией. Потери на излучение обусловлены тем, что при распространении плоской электромагнитной волны в ВС энергия переносится одновременно внутрь и вне него при изгибе. Для поддержания плоского фазового фронта волны при изгибе



оптоволоконна её фазовая скорость на некотором расстоянии от оси должна была бы превысить таковую в материале, что невозможно. Поэтому часть поля, находящаяся на большом расстоянии от центра кривизны, излучается в пространство. Указанный вид потерь преобладает при изгибе одномодовых ВС. В многомодовых ВС, кроме этого вида, имеют место потери за счет конверсии мод. В этом случае при искривлении оптической оси происходит «перекачка» энергии мод низшего порядка в высшие, которые более сильно затухают, чем низшие, из-за большей оптической длины пути. По мере увеличения кривизны все большее число направляемых мод преобразуется в моды оболочки и излучения.

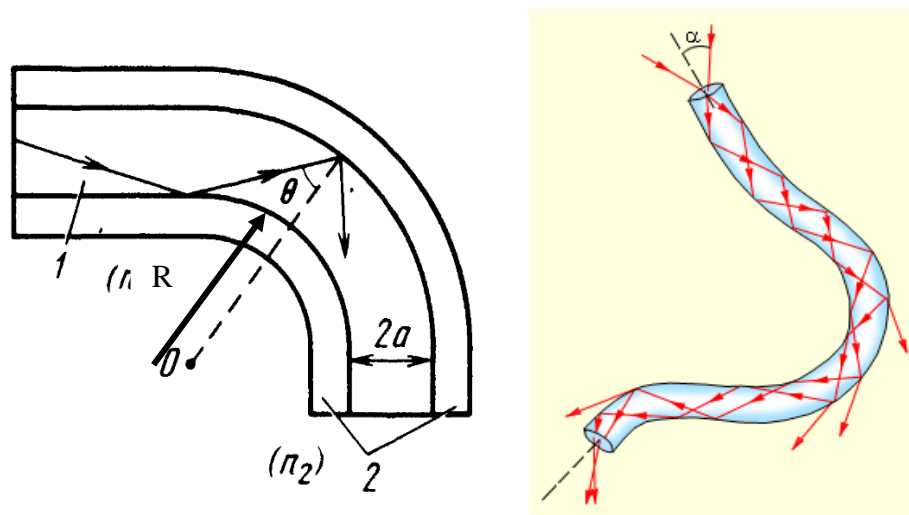


Рис. 2.6. Схематическое изображение изгиба ВС.

1- сердцевина ВС; 2- оболочка ВС

ВС с большей числовой апертурой имеют участок явно выраженной зависимости, полностью смещенный в область меньших радиусов по сравнению с низкоапертурными волокнами. Смещение участка явно выраженной зависимости в область малых радиусов при определенных условиях приводит к тому, что этот участок снизу будет ограничен пределом механической прочности ВС. Это вызовет уменьшение протяженности участка явно выраженной зависимости и снижение чувствительности к изгибу. Светопотери высокоапертурных волокон при одинаковых значениях радиусов кривизны меньше, чем для низкоапертурных.

В экспериментах использовалось многомодовое волокно. Выбор объясняется тем, что зависимость коэффициента ослабления от радиуса изгиба у одномодового ВС носит осциллирующий характер (рис. 2.7.), связанный с процессом туннелирования излучения из сердцевины в оболочку [1, 105]. Осцилляции могут затруднить регистрацию механических воздействий на ВС при изгибе. Для многомодового ВС осцилляции мод накладываются друг на друга и поэтому возможно сглаживание осциллирующей зависимости и получение более монотонного её вида.

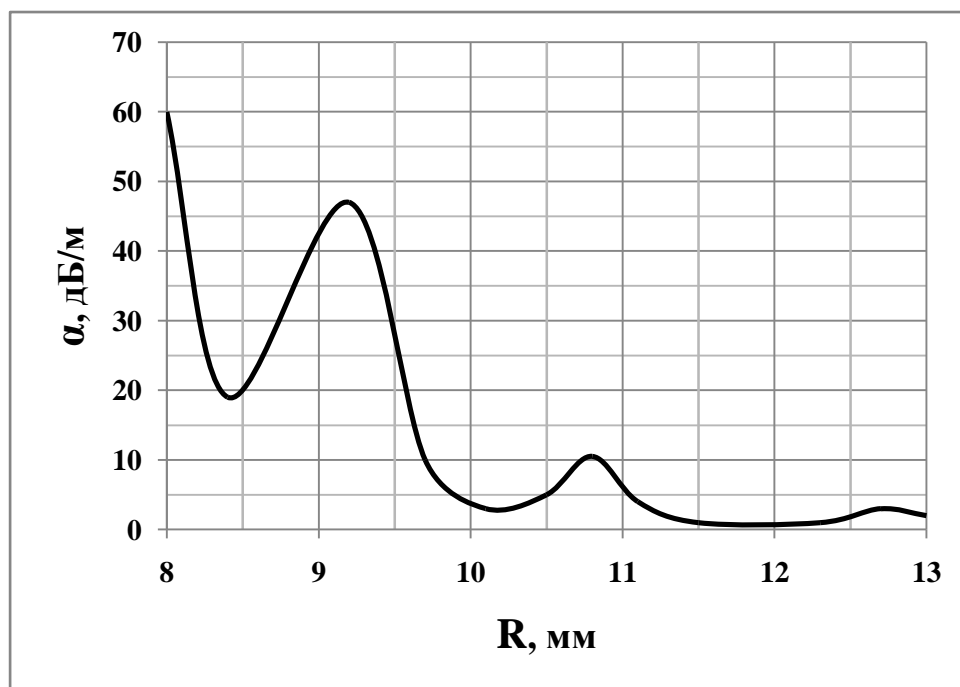


Рис. 2.7. Зависимость коэффициента ослабления мощности от радиуса изгиба стандартного одномодового волоконного световода

ВС на рисунке 2.6. изгибается с радиусом кривизны  $R$ , несколько большим значения критического радиуса изгиба, при котором все излучение покидает сердцевину и выходит в оболочку. Эффективность преобразования мощности [1, 106, 107] пропорциональна отношению числа мод, распространяющихся в оболочке, к полному числу мод и может быть рассчитана по формуле:

$$\frac{N_c}{N} = 1 - 2 \left( \frac{n_1}{n_2} \right) \frac{a}{R \theta_m^2}, \quad (2.2.2)$$

где  $N$  – число мод в световоде;  $N_c$  – число мод в оболочке;  $R$  – радиус изгиба световода;  $\theta_m \approx \arcsin(n_2 / n_1)$ ,  $a$  – радиус сердцевины.

Из (2.2.2) видно, что изменение пропускания оптического канала, возникающего вследствие изменения значения радиуса кривизны изгиба ВС, может быть положено в основу принципа работы амплитудных ВОД, в которых данное изменение обусловлено перемещением, давлением, температурой, усилием и т.д.

Для исследования влияния деформаций изгиба на пропускание оптического волокна была собрана экспериментальная установка, которая представлена на рисунке 2.8 (а, б) вместе со схемой. Оптическое излучение полупроводникового лазера – 1 с длиной волны 650нм вводится в многомодовый ВС – 4, ВС наматывается на цилиндрические стержни – 3 разных радиусов. Получаемый оптический сигнал поступает в цифровой измеритель оптической мощности – 2. Мощность излучения на выходе ВС, указанного на схеме рисунка 2.8 (а), при его изгибе описывается выражением [1, 107-109]:

$$P(z, R) = P(0) \exp([- \alpha(R)]z), \quad (2.2.3)$$

где  $R$  – радиус изгиба ВС;  $z$  – длина изогнутого участка (при одном витке равна длине окружности  $2\pi R$ );  $P(0)$  – исходная мощность направляемого излучения;  $\alpha(R)$  – коэффициент ослабления излучения в изогнутом ВС.

Результаты измерений для зависимости коэффициента ослабления мощности при изгибе ВС от радиуса изгиба приведены на рисунке 2.9. Измерения проводились для 3-х разных многомодовых волокон (рис. 2.10). Для каждого радиуса мощность на выходе ВС измерялась 5 раз, а окончательный результат получали усреднением этих измерений. Данные зависимости подтверждают тенденцию нарушения светопропускания изгибаемого световода при уменьшении его радиуса изгиба. Повысить чувствительность изгибных преобразователей

можно следующими способами, основанными на повышении чувствительности элементарных преобразований [1, 106-120]: увеличением мощности излучения, вводимого в ВС; применением градиентных световодов и ВС с большим диаметром сердцевины и сравнительно низкой числовой апертурой при возбуждении их изотропным источником. Влияние диаграммы направленности источника излучения можно объяснить тем, что при изгибе энергия мод высших порядков переходит в вытекающие из ВС моды излучения.

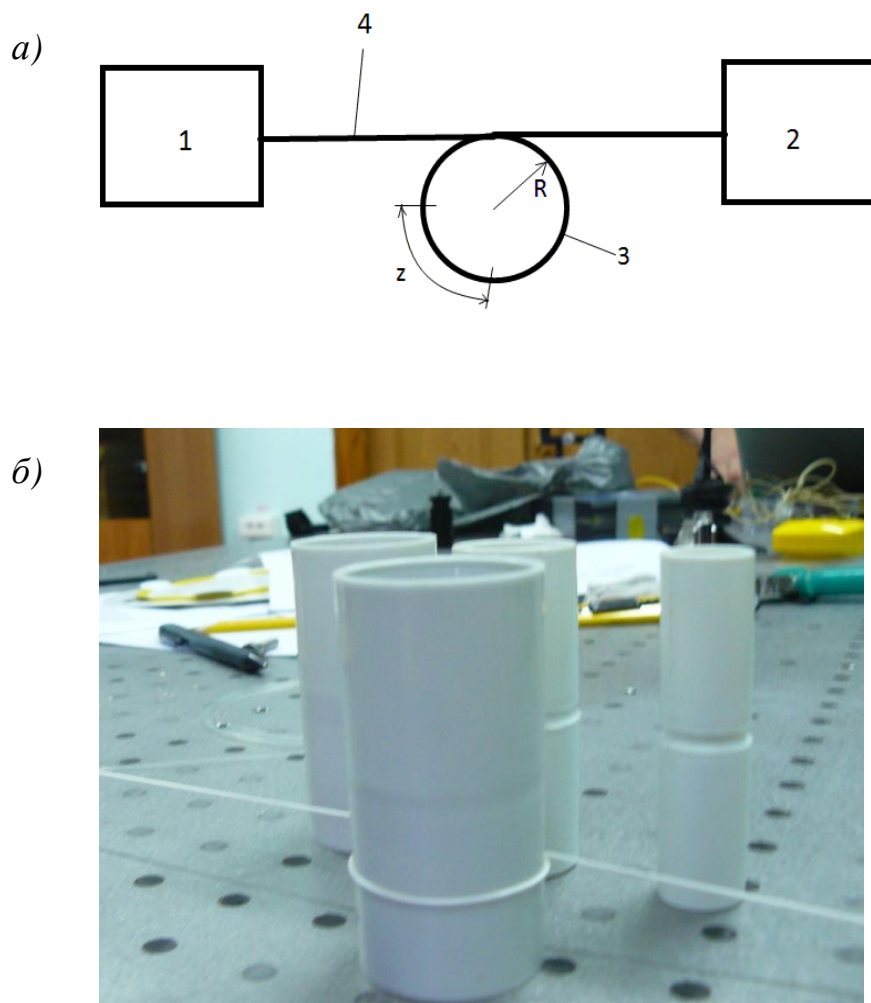


Рис. 2.8. *а)* Установка для экспериментального исследования влияния радиуса изгиба на пропускание оптического волокна (схема установки): 1- излучатель, 2- цифровой измеритель оптической мощности, 3 - набора цилиндрических стержней разных диаметров, 4 – ВС;  
*б)* Установка для экспериментального исследования влияния радиуса изгиба на пропускание оптического волокна (фотография цилиндрического стержня с намотанным на нём ВС)

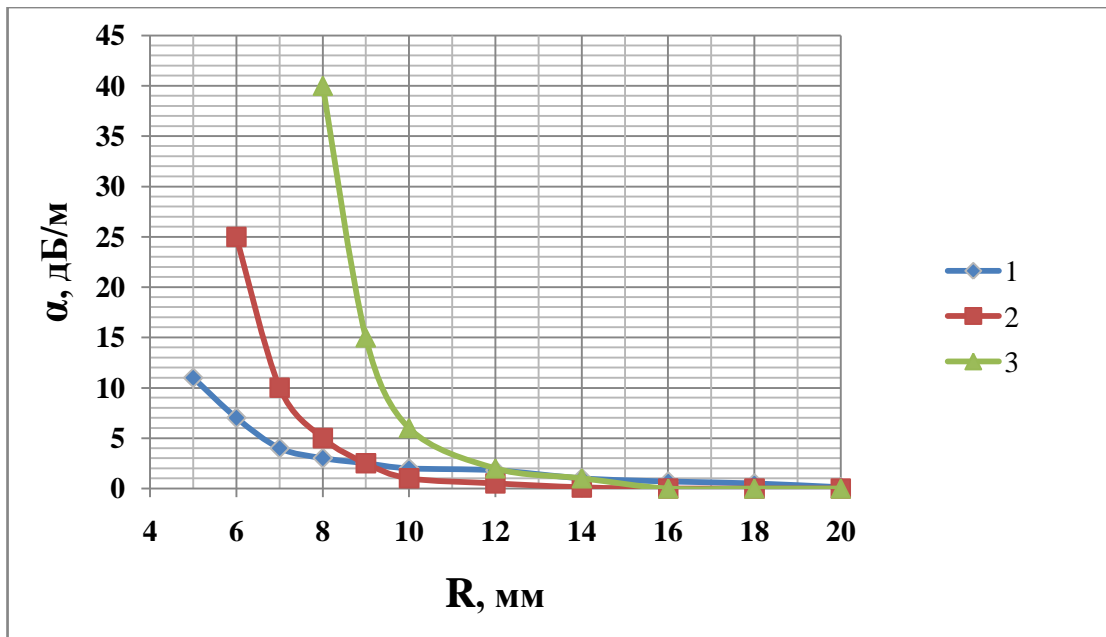


Рис. 2.9. Зависимость коэффициента ослабления направляемого излучения от радиуса изгиба. 1 – стандартный многомодовый патчкорд 50мкм/125мкм, 2 – многомод без буферного покрытия 50мкм/125мкм NA=0,2, 3 – многомод 50мкм/125мкм NA=0,215

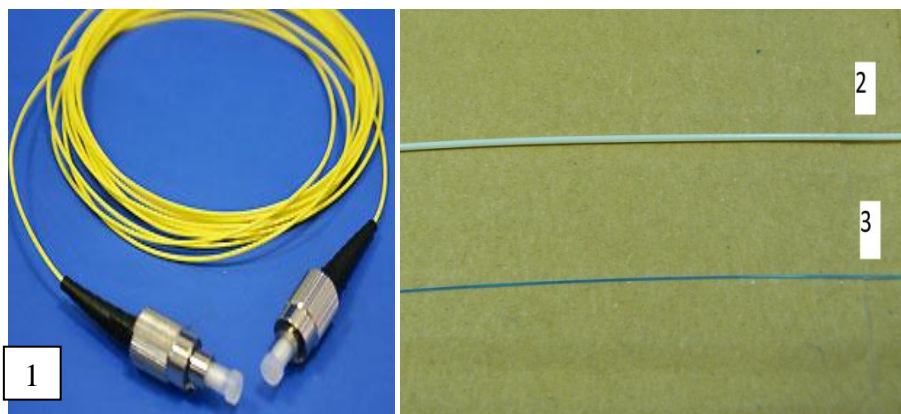


Рис. 2.10. Применяемые в эксперименте многомодовые ВС. 1 – стандартный многомодовый патчкорд 50мкм/125мкм; 2 – многомод 50мкм/125мкм NA=0,215; 3 – многомод без буферного покрытия 50мкм/125мкм NA=0,2

А в случае направленного излучения парциальные мощности мод высших порядков меньше, чем низших, поэтому при уменьшении радиуса кривизны в области относительно больших их значений потери незначительны, а по мере увеличения кривизны довольно резко возрастают. В случае же изотропного возбуждения возрастание потерь происходит монотонно. Для ЧЭ линии измерительной сети, находящейся в слое грунта на глубине порядка 5см от

поверхности для регистрации внешнего воздействия (давления) на верхний слой грунта необходимо обладать высокой чувствительностью, чтобы с достаточной точностью определить прогиб почвы на значение порядка 1мм. Повысить чувствительность такой измерительной системы также можно, выполнив ЧЭ в виде многовитковой конструкции. Такая конструкция позволяет повысить чувствительность к изгибу в 2-5 раз, но при этом происходит увеличение жесткости и усложнение конструкции. В результате измерений были получены зависимости мощности на выходе ВС от числа витков для различных радиусов изгиба, представленные на рисунке 2.11.

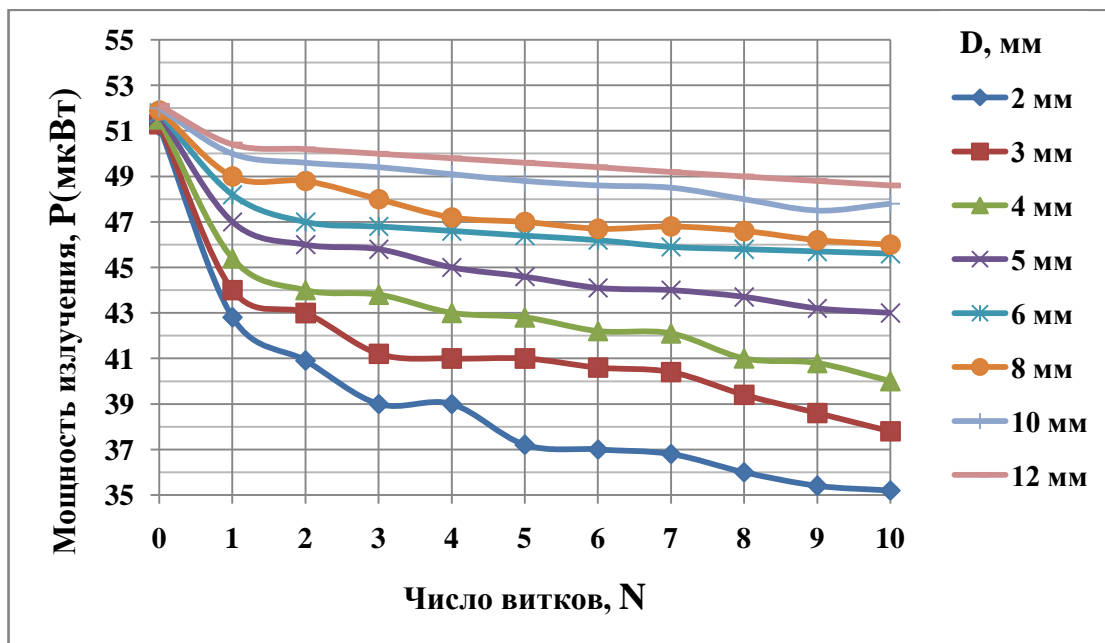


Рис.2.11. Зависимость пропускания от числа витков волновода при постоянных радиусах изгиба для многомодового ВС

Из рисунка 2.11 видно, что ЧЭ целесообразно выполнять в виде 2-4 витков ВС. Такое количество витков определяется тем, что моды при нарушенном изгибом условия распространения практически полностью покидают ВС. Поведение излучения на прямолинейном участке за изгибом оптоволоконного кабеля проанализировано в [1, 111-121]. Установлено, что около 80% излучения, вышедшего на изгибе в оболочку, возвращаются обратно в сердцевину на прямолинейном участке за изгибом ВС. Применение же многовитковой конструкции решает эту проблему.

В результате проведенных исследований было экспериментально подтверждено возникновение потерь мощности при изгибе оптоволокна и возрастание их при уменьшении радиуса изгиба участка ВС. Экспериментально была получена характеристика зависимости коэффициента потерь мощности в ВС от радиуса его изгиба. Описаны методы повышения чувствительности ВС к изгибу и дано их объяснение.

### 2.2.3 Волоконно-оптические датчики на основе одноволоконного многомодового интерферометра

Как показали результаты многочисленных исследований [1, 94, 122-126], волоконные интерферометры позволяют потенциально достичь наибольшей чувствительности для измерительных устройств. Одноволоконный многомодовый интерферометр (ОМИ), изображенный на рисунке 2.12 представляет собой волоконный световод, в котором возбуждается два или более направляемых луча (мод).

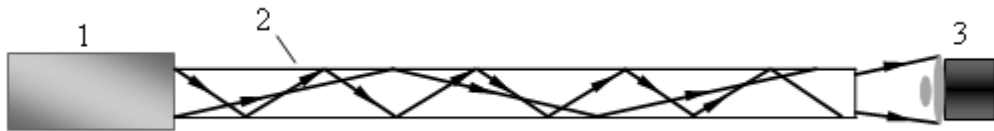


Рис. 2.12. Одноволоконный многомодовый интерферометр.

1 – лазер; 2 – многомодовый волоконный световод ; 3 – фотоприемное устройство

Вследствие того, что эти лучи распространяются в ВС под разными углами, они пройдут разные расстояния, а следовательно, фазовые скорости их распространения также будут различными. Поэтому на выходе из ВС между этими лучами будет возникать разность фаз, которая равна:

$$\Delta\varphi = k\Delta n_{эф} \cdot L, \quad (2.2.4)$$

где  $\Delta n_{эф}$  - разность эффективных показателей преломления разных мод. Как видно из выражения (2.2.4), разность фаз между модами будет изменяться с

изменением длины ВС  $L$ . Вследствие этого будет изменяться и картина интерференции мод на выходе из ОМИ, что можно регистрировать.

Для регистрации внешних воздействий также можно использовать датчики на основе таких интерферометров как интерферометр Маха-Цеднера и интерферометр Фабри-Перо [1, 90, 91, 126-128]. Для распределенной измерительной сети датчики на основе интерферометра Маха-Цеднера обладали бы серьезным недостатком: опорное плечо интерферометра должно быть хорошо изолировано от внешних воздействий, что в случае измерительной сети представляется сложно выполнимой задачей. Интерферометры Фабри-Перо нуждаются в высоко когерентных источниках излучения и требуют использования специальных электронных схем стабилизации, что ведет к усложнению и удорожанию измерительной системы. Преимущество ОМИ по сравнению с интерферометрами Маха-Цеднера и Фабри-Перо заключается в том, что такие устройства позволяют сочетать такие качества, как предельную простоту схемы, которая содержит единственный волоконный световод, работающий в многомодовом режиме, высокие метрологические характеристики, устойчивость к электромагнитным помехам, агрессивным средам и радиации. При этом они имеют достаточно низкую стоимость, что также является очень важным фактором в случае использования большого количества чувствительных элементов в измерительных сетях. Следовательно, использование ОМИ в измерительных сетях имеет определенные перспективы использования для решения задач в той постановке, которая предложена в данной работе [1, 90, 91, 94, 122-128].

Для измерения величины внешнего воздействия на ОМИ используют результат интерференции направляемых лучей или мод одного и того же ВС. Поскольку в ОМИ распространяется большое количество мод, излучение, выходящее из него представляет сложный оптический сигнал. Этот сигнал формируется в результате интерференции большого числа мод ВС, у каждой из которых есть свое пространственное распределение и своя фазовая задержка. На выходе ОМИ, вследствие интерференции мод, можно наблюдать



соответствующую спекловую картину, изображенную на рисунке 2.13. Спекл-картина является результатом статического усреднения амплитудно-фазовых полей направляемых мод, амплитуда и фаза которых может изменяться случайным образом по мере распространения в ВС.

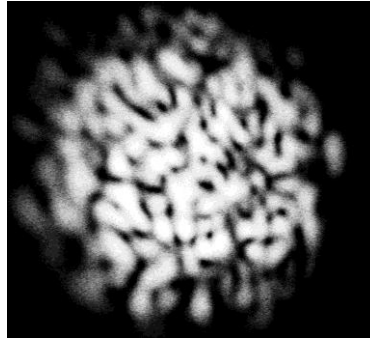


Рис.2.13. Спекловая картинка на выходе из ОМИ

Эти случайные изменения амплитуды и фазы возникают из-за неоднородностей в материале ВС, возникающих при его изготовлении, а также в результате многократного переотражения световых лучей. Вследствие случайности фазовых сдвигов мод и изменений амплитуды, интенсивность излучения будет распределяться так же случайным образом. Таким образом, если на ВС в ОМИ оказывать какое-либо внешнее воздействие, то это приведет к дополнительным фазовым сдвигам, что следует из (2.2.4), вследствие чего будет наблюдаться пространственное перераспределение структуры спекл-картины. В результате соответствующего анализа и обработки полученной картины на выходе, становится возможным определить характер внешнего воздействия на ОМИ [1, 90, 91, 94, 122-128].

#### **2.2.4 Модель конструкции чувствительного элемента для волоконно-оптической измерительной линии на основе ОМИ**

В результате проведенных исследований была определена модель конструкции ЧЭ на основе одноволоконного многомодового интерферометра для реализации РВОИС в предложенной нами системе ИРИИС, которая показана на

рисунке 2.14. и обеспечивает необходимую чувствительность, в сочетании с простотой ЧЭ и возможностью построения протяженных распределенных измерительных линий.

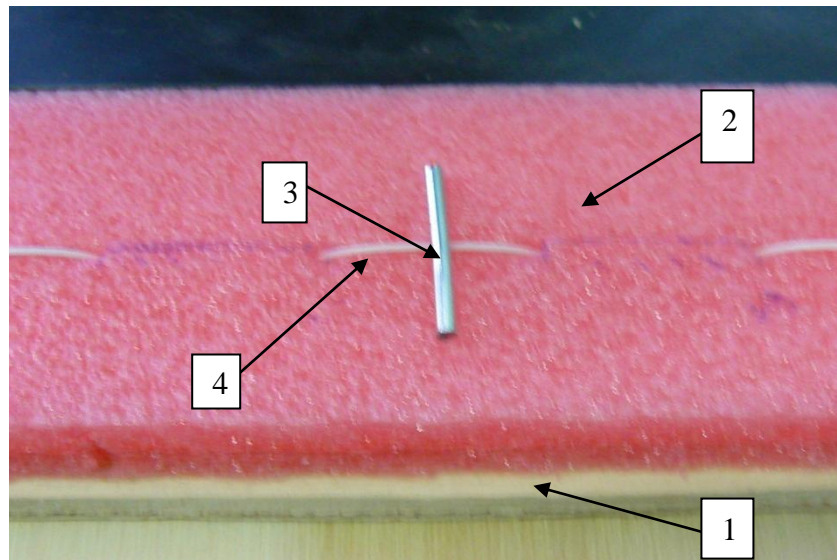


Рис. 2.14. Чувствительный элемент на основе одноволоконного многомодового интерферометра.

1- твердая подложка, 2- упругая среда, 3-вдавливающий стержень, 4 - волоконный световод

Предложенная модель ЧЭ состоит из твердой подложки – 1 (фанера), на которую приклеен упругий материал (упругая среда) – 2, в качестве которого был использован поролон. Упругая среда была прошита многомодовым оптоволоконном – 4, как показано на рисунке 2.15. Оптоволоконно вдавливалось в нее с помощью металлических цилиндрических стержней – 3, обеспечивающих деформацию волокна, а, следовательно, и изменение фаз мод, влияющих на вид спекл-картины. При прекращении надавливания волокно приводилось в исходное состояние, так как упругая среда возвращала свою форму.

Применение упругой среды в ЧЭ решает несколько проблем:

- возвращение ЧЭ в исходное состояние;
- компенсация изменения длины чувствительного участка, которое может привести к чрезмерному натяжению ВС и в конечном итоге к обрыву волокна;

- подбирая модуль Юнга упругой среды, можно добиться необходимого коэффициента преобразования давления на упругую среду в изменение ее толщины, а, следовательно, и величину деформации ВС;
- недопущение чрезмерного прогиба волокна, который может привести к излому, а, следовательно, и к выходу из строя ЧЭ.

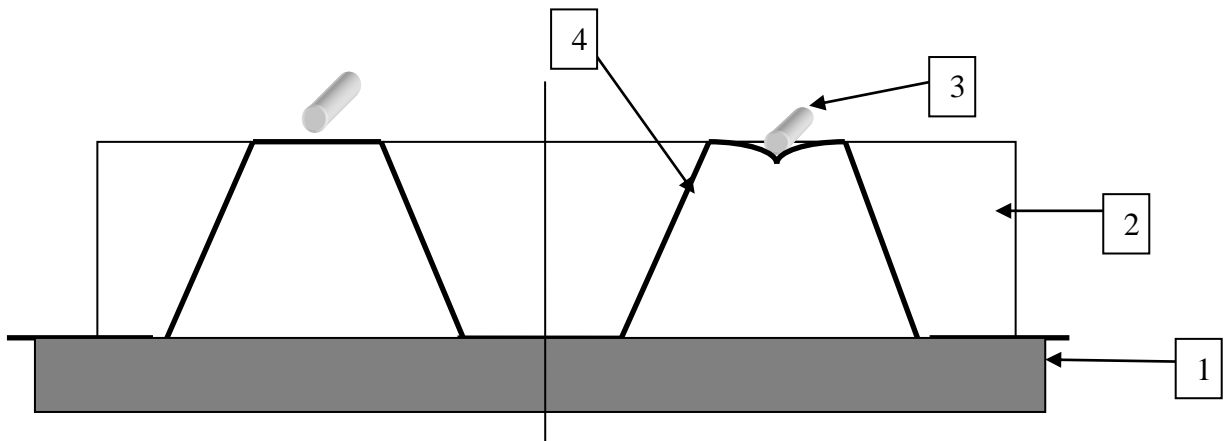


Рис. 2.15. Чувствительный элемент в разрезе.

Слева – исходное состояние, справа – деформация ВС. 1 – твердая подложка, 2 – упругая среда, 3 – вдавливающий стержень, 4 – волоконный световод

В ЧЭ применялось стандартное многомодовое волокно с буферным покрытием, со ступенчатым показателем профиля преломления, диаметр сердцевины 50мкм, диаметр оболочки 125мкм  $NA=0,215$ . В ВС подавалось излучение с полупроводникового лазера с рабочей длиной волны 650нм. Для данного устройства были разработаны принципы и методы обработки выходного оптического сигнала с ЧЭ.

### **2.3 КОРРЕЛЯЦИОННАЯ ОБРАБОТКА СИГНАЛА НА ВЫХОДЕ ОДНОВОЛОКОННОГО МНОГОМОДОВОГО ИНТЕРФЕРОМЕТРА ПРИ МАКРОИЗГИБЕ**

Корреляционную обработку сигналов применяли для измерения величины удлинения ВС в составе одноволоконных многомодовых интерферометров при помощи приборов с зарядовой связью для оптических методов измерения

деформационных воздействий на ОМИ. Результаты этих исследований описаны в работах [129-133]. Но в них не рассматривалась возможность использования подобной обработки при макроизгибе ВС.

Зависимость коэффициента корреляции интерференционных полей, сформированных ОМИ, в зависимости от максимального значения дополнительной разности фаз между модами интерферометра показана в работах [131, 132] формулой:

$$\rho_{12} = \frac{\sin^2\left(\frac{\Delta\Psi_{\max}}{2}\right)}{\left(\frac{\Delta\Psi_{\max}}{2}\right)^2}, \quad (2.3.1)$$

В приближении  $LP$ -мод изменение фазы моды  $LP_{pq}$ , вызванное деформацией ВС, имеет вид [131-133]:

$$\Delta\Psi_{pq} = kn_{pq} \frac{dl}{dZ} \Delta Z + kl \frac{dn_{pq}}{dZ} \Delta Z, \quad (2.3.2)$$

где  $n_{pq} = (\beta_{pq} / k)$  - эффективный показатель преломления для моды  $LP_{pq}$ ,

$l$  - длина световода,  $\beta_{pq}$  - постоянная распространения этой моды,  $\Delta Z$  - амплитуда внешнего деформационного воздействия,  $k = 2\pi / \lambda$ ,  $\lambda$  - длина волны излучения. При этом предполагается, что вариация эффективного показателя преломления  $LP$ -моды вследствие деформации ВС может быть обусловлена аксиальным растяжением световода и его малыми макроскопическими изгибами. При этом возможность микроизгибов, равно как и других факторов, влияющих на амплитуду мод, не рассматривается.

Известно, что при макроизгибе аксиальное растяжение ВС мало, а изменение показателя преломления одинаково для всех мод, поэтому этими причинами существенное изменение корреляции (2.3.1) не вызвано согласно формуле (2.3.2) [131-135]. Потери мощности излучения, возникающие на изгибе, значительны, поэтому, если учитывать изменение интенсивности и, исходя из этого, определять изменение коэффициента корреляции, то можно будет установить зависимость его значения от глубины прогиба ВС.

### 2.3.1 Обработка выходного оптического сигнала с чувствительного элемента на основе ОМИ

Для захвата выходного сигнала в виде спекл-картины с чувствительного элемента датчика была собрана установка, представленная на рисунке 2.16.

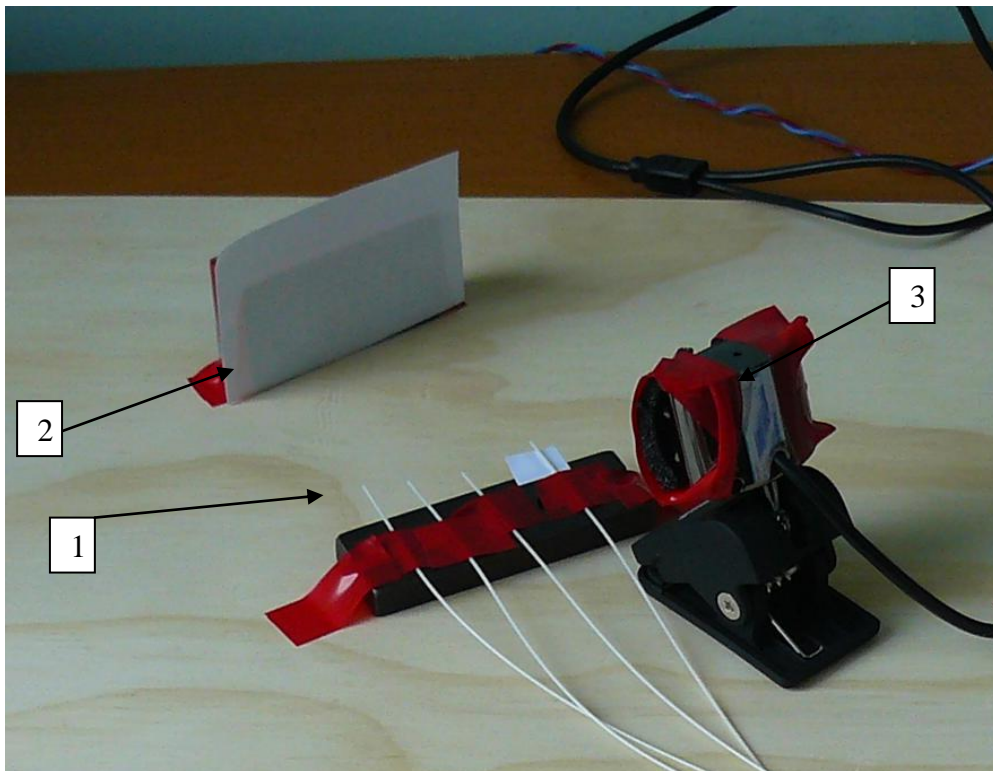


Рис.2.16. Установка для захвата спекл-картины на выходе чувствительного элемента:

1 – торец волоконного световода, 2 – экран, 3 – WEB-камера

Выходной сигнал в виде спекл-картины с торца волоконного световода – 1 проецировался на экран – 2, и спекл-картина снималась с помощью WEB-камеры (ПЗС) – 3. WEB-камера была подключена к компьютеру, данные с нее обрабатывались с помощью специально написанной программы. При проведении экспериментов установка закрывалась непрозрачным кожухом, для предотвращения влияния дневного света на регистрацию спекл-картины. При регистрации изображения спекл-картины с использованием ПЗС матрицы происходит сохранение указанного изображения в памяти ЭВМ в виде матрицы значений  $I(x,y)$ , значение каждого элемента которой представляет собой результат

аналого-цифрового преобразования величины световой интенсивности, регистрируемой элементом ПЗС матрицы с координатами  $x$  и  $y$ .

Для корректного захвата спекл-картины ПЗС необходимо соблюсти несколько условий: обеспечить линейный режим ПЗС и оптимальный размер захватываемой картинки.

Приборы с зарядовой связью обладают очень большим, но ограниченным диапазоном линейности. При превышении некоторой пороговой интенсивности  $I_H$  светового потока наступает переполнение потенциальной ямы пикселей ПЗС фотоэлектронами и величина электрического сигнала на выходе прибора больше не растёт (рис. 2.17) [131, 132]. Из-за очень большой дисперсии интенсивности света в спекл-картине ее значения для некоторых локальных участков картины могут превышать пороговое значение  $I_H$ . Для этих участков будет нарушена прямая пропорциональность между измеряемыми ПЗС значениями и интенсивностью света. Следствием этого будет дополнительная погрешность при измерении величины коэффициента корреляции спекл-сигналов [131, 132].

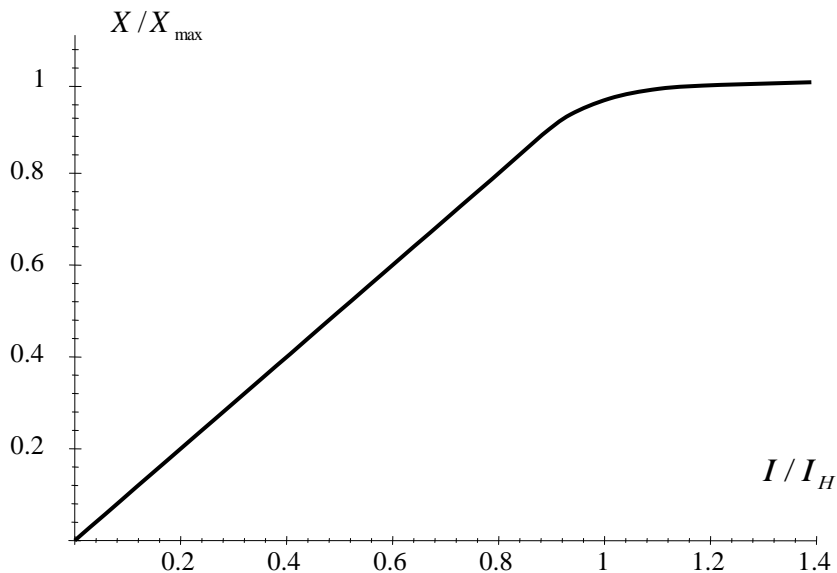


Рис. 2.17. Зависимость выходного сигнала элементарной светочувствительной ячейки (пикселя) ПЗС от интенсивности направляемого на нее светового потока

Существует оптимальная величина радиуса захвата спекл-картины  $R_{\text{опт}}$  (рис. 2.18). При  $R < R_{\text{опт}}$  коэффициент корреляции может принимать отрицательные значения. Это может быть объяснено недостаточным количеством зарегистрированных спеклов для обеспечения корреляционной обработки данных и коррелированной перестройки самих спеклов при малых размерах выделенного участка интерференционной картины для обработки [130-135].

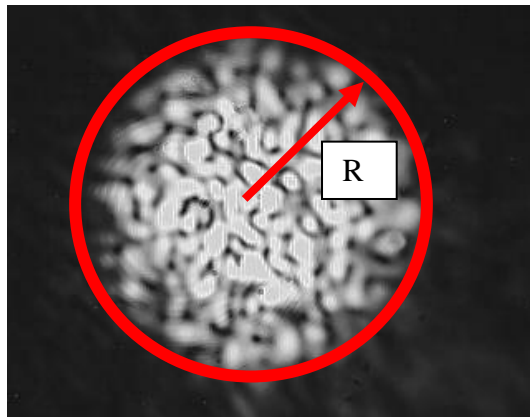


Рис. 2.18. Радиус захвата спекл-картинки

Действительно, если в пределах такого участка регистрируется единственное световое пятно, которое вследствие коррелированной перестройки постепенно перемещается из одной в другую его часть, то конечное изображение участка будет антикоррелировать с начальным изображением как видно на рисунке 2.19 [131, 132].

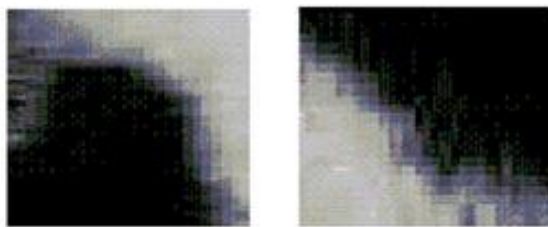


Рис. 2.19. Анतिकорреляция изображений

Когда же величина радиуса картины близка к  $R_{\text{опт}}$ , регистрируется достаточно большое количество спеклов, взаимное перемещение которых некоррелировано.

В этом случае, в процессе растяжения или изгиба ВС коэффициент корреляции снижается до нуля. Если же  $R > R_{\text{опт}}$ , ПЗС матрица регистрирует внешние по отношению спекл-картине участки, яркость которых не зависит от деформации ОМИ. Вследствие этого коэффициент корреляции уменьшается до некоторого положительного значения [131, 132].

Обработка информации, поступающей с ПЗС матрицы, была выполнена по программе в среде MATLAB. Алгоритм обработки выходного оптического сигнала с чувствительного элемента на основе ОМИ при макроизгибе состоит в следующем:

При запуске программы ПЗС матрица видеокамеры регистрирует опорное изображение спекл-картины, соответствующее начальному состоянию ВС в ЧЭ. В памяти компьютера опорное изображение представляется в виде двухмерного массива  $I^{\text{оп}}(x,y)$ , значение элементов массива равно интенсивности падающего света на точку в плоскости ПЗС матрицы с координатами  $x$  и  $y$  и остается в памяти в течение всего времени измерения. Деформация световода вследствие внешних воздействий приводит к изменению спеклов в регистрируемой ПЗС матрицей картине, изменившееся распределение интенсивности записывается в массив  $I(x,y)$ . Сравнение спекл-сигналов до и после деформационного воздействия на ВС производится корреляционным способом. В качестве меры сравнения использовали парный рядный коэффициент корреляции [136] в виде формулы:

$$\rho_{12} = \frac{\sum_{x,y} \left( I^{\text{оп}}(x,y) - \overline{I^{\text{оп}}(x,y)} \right) \left( I(x,y) - \overline{I(x,y)} \right)}{\sqrt{\left( \sum_{x,y} \left( I^{\text{оп}}(x,y) - \overline{I^{\text{оп}}(x,y)} \right)^2 \right) \left( \sum_{x,y} \left( I(x,y) - \overline{I(x,y)} \right)^2 \right)}}, \quad (2.3.3)$$

Воздействие шумов системы (пичковый режим работы лазера, наводки, шумы ПЗС матрицы и т.д.) проявляется в том, что матрицы значений интенсивности света в памяти компьютера при одних и тех же внешних условиях



с течением времени будут отличаться, а, следовательно, коэффициент корреляции нестабилен (рис. 2.20).

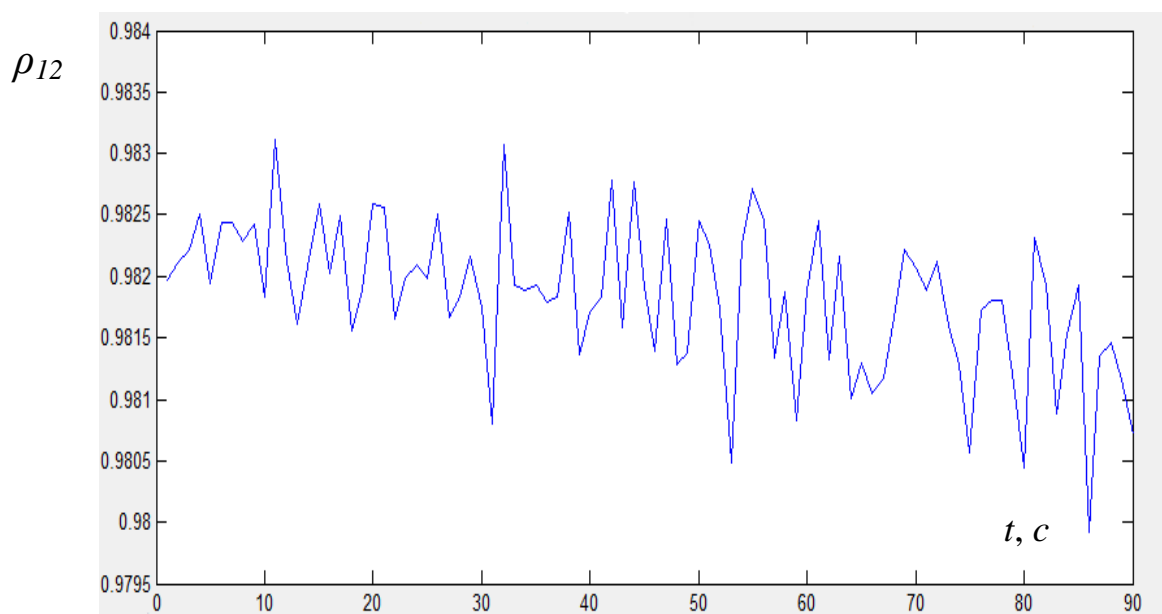


Рис. 2.20. Влияние шумов системы на изменение значения коэффициента корреляции

Из рисунка 2.20 видно, что разница между максимальным и минимальным значением коэффициента корреляции имеет значение 0,003, что на 3 порядка меньше максимального значения корреляции (т.е.  $\rho_{12} = 1$ ), и как следствие шумы не будут вносить существенный вклад в результаты измерений, ошибка определения  $\rho_{12}$  составит до 1% от самого значения.

Для определения зависимости изменения значения коэффициента корреляции от величины деформации ВС была собрана установка, показанная на рисунке 2.21. ВС – 1 вдавливался в упругую среду – 2 посредством вдавливающего стержня – 3. Величина вдавливания регулировалась подвижкой – 4. За начало отсчета принято положение стержня, при котором волокно не деформировано, а, следовательно,  $\rho_{12} = 1$ . Деформация ВС посредством стержня приводит к изменению коэффициента корреляции.

График зависимости коэффициента корреляции от глубины прогиба при вдавливании ВС представлен на рисунке 2.22. Как видно из графика, ЧЭ на

основе ОМИ обладает явно выраженным линейным участком зависимости и заметной чувствительностью к воздействию, вызывающего прогиб ВС.

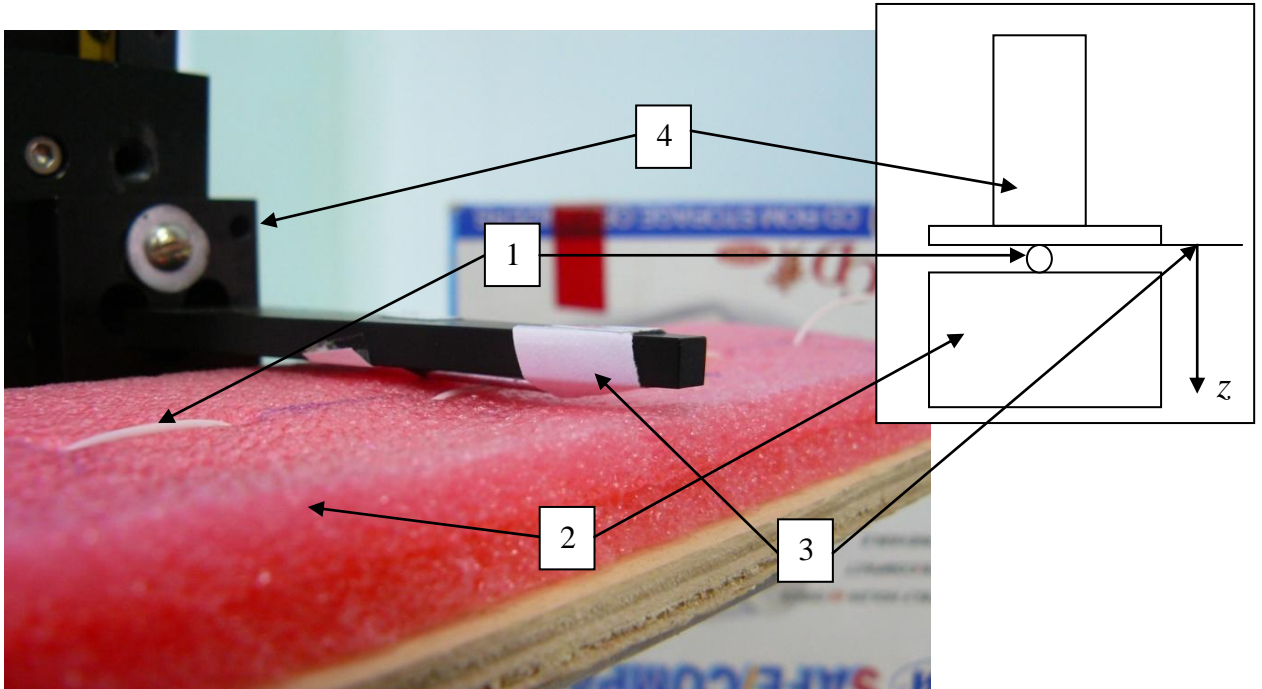


Рис. 2.21. Установка для снятия значения коэффициента корреляции от величины деформации волоконного световода:

1 – волоконный световод, 2 – упругая среда, 3 – вдавливающий стержень, 4 – подвижка

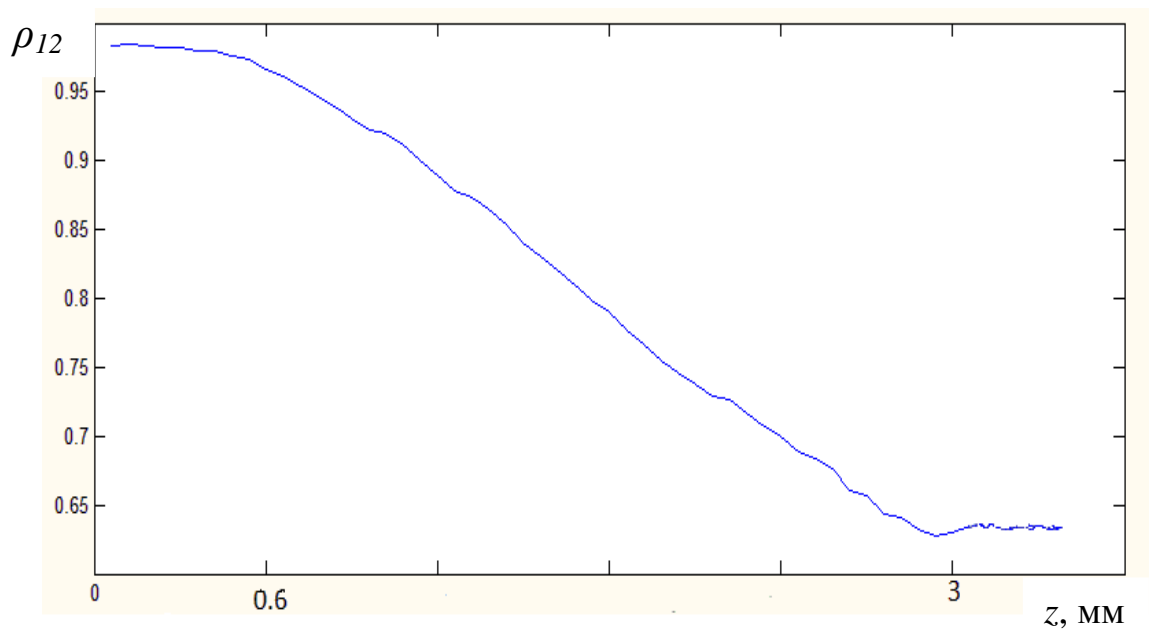


Рис. 2.22. Зависимость коэффициента корреляции от глубины вдавливания ВС

Проанализировав график можно сделать вывод, что предложенный ЧЭ имеет линейный диапазон измеряемого поперечного воздействия в интервале 0,6-3мм с чувствительностью  $0,1\text{мм}^{-1}$  и погрешностью 1%, обусловленной шумами. Причем, зная величину прогиба и характеристики упругой среды, можно восстановить значение массы тела, действующей на ЧЭ.

Также следует отметить, что при вдавливании может наступить момент отсечки, вследствие чего дальнейшее вдавливание ВС в упругую среду не приведет к уменьшению коэффициента корреляции (область после 3мм на рисунке). Это недопустимо при использовании ЧЭ в составе интегрирующей измерительной линии, при анализе сигнала на выходе которой, необходимо получить суммарное значение воздействия на чувствительные элементы. Для выхода из этой ситуации при измерении, можно перезаписывать в памяти компьютера опорный сигнал каждый раз, когда коэффициент корреляции уменьшается до определенного порогового значения, как это иллюстрирует рисунок 2.23, на котором представлены зависимости  $\rho_{12}(z)$  до (кривая 1) и после (кривая 2) процедуры перезаписи. При этом значение глубины прогиба при вдавливании ВС, найденное после последней процедуры перезаписи, суммируется со значением, полученным до проведения этой процедуры. Применяя указанный алгоритм, можно проводить количественные измерения суммарного воздействия на ЧЭ в составе интегрирующей измерительной линии.

Экспериментально полученное изменение коэффициента корреляции при последовательной деформации трех ЧЭ, входящих в измерительную линию, с применением процедуры перезаписи опорного сигнала демонстрирует рисунок 2.24. Подобная процедура успешно использовалась в работах [131, 132] при измерении удлинения маломодового ВС. Пользуясь описанным методом, можно проводить измерения сколь угодно больших суммарных деформаций, а по знаку изменения средних значений интенсивностей последовательности опорных и текущих спекл-сигналов можно определить, что происходило при этом (усиление или ослабление воздействия на измерительную линию).

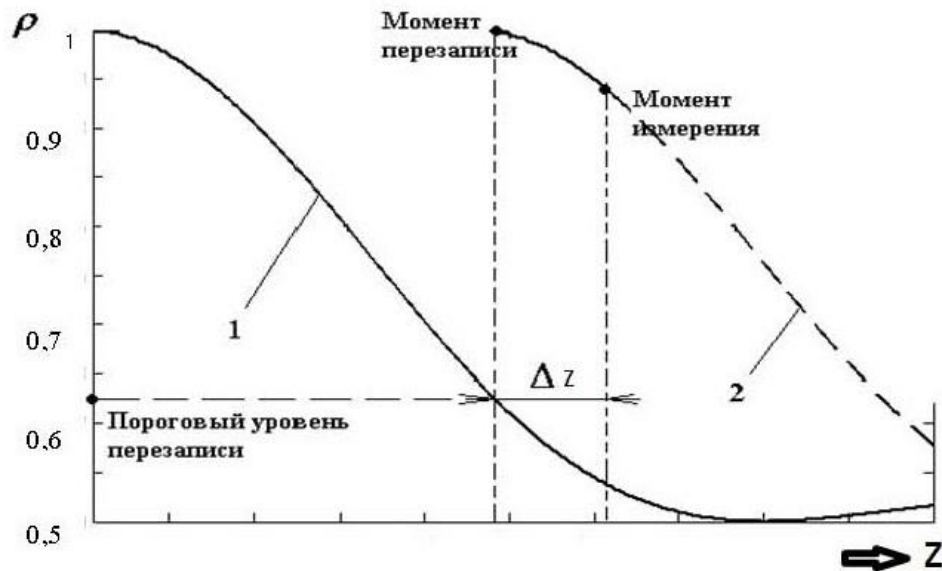


Рис. 2.23. Зависимости коэффициента корреляции от деформации ВС, при перезаписи опорного изображения:

1 – зависимость  $\rho_{12}(z)$  до момента перезаписи опорного изображения, 2 – зависимость  $\rho_{12}(z)$  после момента перезаписи опорного изображения

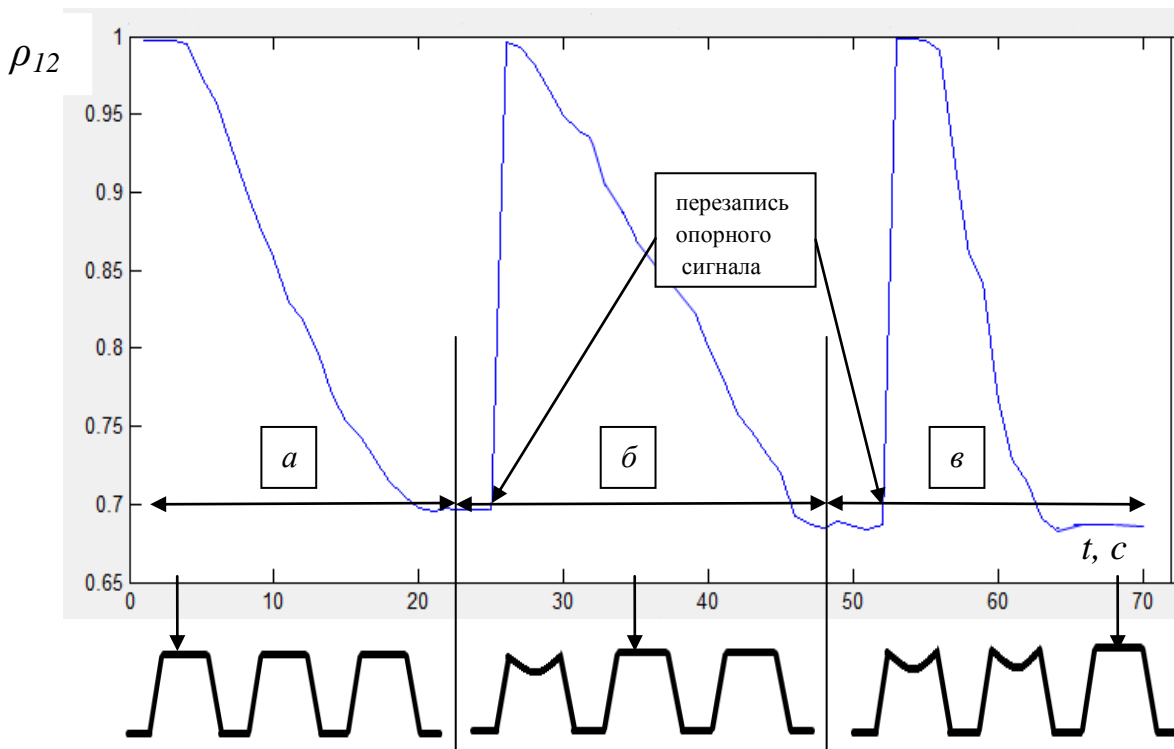


Рис. 2.24. Экспериментально полученное изменение коэффициента корреляции при последовательной деформации трех ЧЭ, входящих в измерительную линию с применением процедуры перезаписи опорного сигнала. *a* – воздействие на первый ЧЭ, *б* – воздействие на второй ЧЭ, *в* – воздействие на третий ЧЭ

Быстродействие метода определяется использованием высокоскоростных специализированных видеокамер, которые способны регистрировать с частотой кадров до нескольких тысяч Гц и скоростью самой компьютерной обработки данных, вследствие чего возрастание быстродействия метода зависит от производительности компьютеров или оптимизации процесса вычислений.

## **2.4 РАЗРАБОТКА ВОЛОКОННО-ОПТИЧЕСКОЙ ИЗМЕРИТЕЛЬНОЙ СИСТЕМЫ ОХРАНЫ ПЕРИМЕТРА**

В качестве датчиков для систем охраны периметра [142-146] применимы те же волоконно-оптические кабели [138-141], которые используются обычно для передачи информации. Деформация оптического волокна изменяет его оптические параметры и, как следствие, характеристики проходящего через волокно излучения. Оптическое волокно в общем случае представляет собой коаксиальный световод. Свет распространяется вдоль центральной части (сердцевины) кабеля. К сердцевине волокна прилегает прозрачная оболочка, которая обладает меньшим показателем преломления, чем сердцевина. Свет, распространяющийся под углом к оси ВС, отражается от границы раздела между сердцевинной и оболочкой и концентрируется в центральной части волокна. Внешнее непрозрачное покрытие служит для механической защиты кабеля [147-152].

В качестве источника излучения обычно используются миниатюрные полупроводниковые лазеры или светодиоды [109-120]. На выходе кабеля излучение регистрируется фотоприемником, который преобразует оптический сигнал в форму электрического. При деформациях волокна изменяются условия внутреннего отражения, в результате чего претерпевают изменения фазовые и пространственные характеристики луча на выходе кабеля. Эти изменения регистрируются фотоприемником и обрабатываются анализатором сигналов [105-108]. Полупроводниковый лазер обычно генерирует несколько десятков близких по частоте мод (спектральных линий) с определенным распределением энергии по спектру излучения. Если многомодовый оптоволоконный кабель подвергается

механическим воздействиям, то на его выходе регистрируемый приемником спектр излучения меняется, что позволяет детектировать деформации кабеля.

Для соединения измерительных линий с излучателем и системой обработки возникла необходимость воспользоваться различными способами соединения волоконных световодов [138-141, 147-152]. При изготовлении макета ВОИС были использованы стандартные технологии сращивания ВС. Реализация таких технологий заключалась в подготовке ВС и выполнении соединения методами сварки или с помощью специальных соединителей.

#### **2.4.1 Принципы создания чувствительного элемента на основе многомодового волоконного световода**

Измерительная линия ИС состоит из отдельных чувствительных элементов. Чувствительный элемент создаётся на основе изгиба многомодового ВС. (см. раздел 2.2.2.)

Принцип работы чувствительного элемента подробно изложен выше и заключается в том, что при изгибе волоконного световода изменяется мощность оптического излучения, распространяющегося в ВС, что регистрируется высокочувствительным фотоприемником.

Для амплитудных датчиков большое значение имеет стабильность мощности источника. Перед началом эксперимента полупроводниковый лазер с рабочей длиной волны 650нм, подающий излучение в ВС, был проверен на стабильность. Исследования показали, что после 15 мин излучение стабилизируется (рис. 2.25).

Для создания чувствительного элемента была выбрана конструкция, представленная на рисунке 2.26. В разработанной модели ЧЭ в качестве элемента, преобразующего воздействие в изгиб волоконного световода, была выбрана конструкция, состоящая из трёх металлических стержней, соединённых между собой [80, 81]. При использовании трёх отдельных стержней необходимо вводить элемент позволяющий стержню 2 после воздействия возвращаться в исходное положение, что приводит к усложнению конструкции. При разработке

чувствительного элемента с оптимальными параметрами использовалось три типа металлических стержней с различными диаметрами: 0,5 мм, 1,5 мм, 2 мм. Для определения оптимального диаметра стержней использовалась установка, представленная на рисунке 2.27.

Важным параметром чувствительного элемента является расстояние между опорными стержнями (рис. 2.26). Для определения данного параметра проводились измерения для пяти различных расстояний ( $L=8$  мм,  $L=10$  мм,  $L=20$  мм,  $L=30$  мм,  $L=36$  мм) между опорными стержнями.

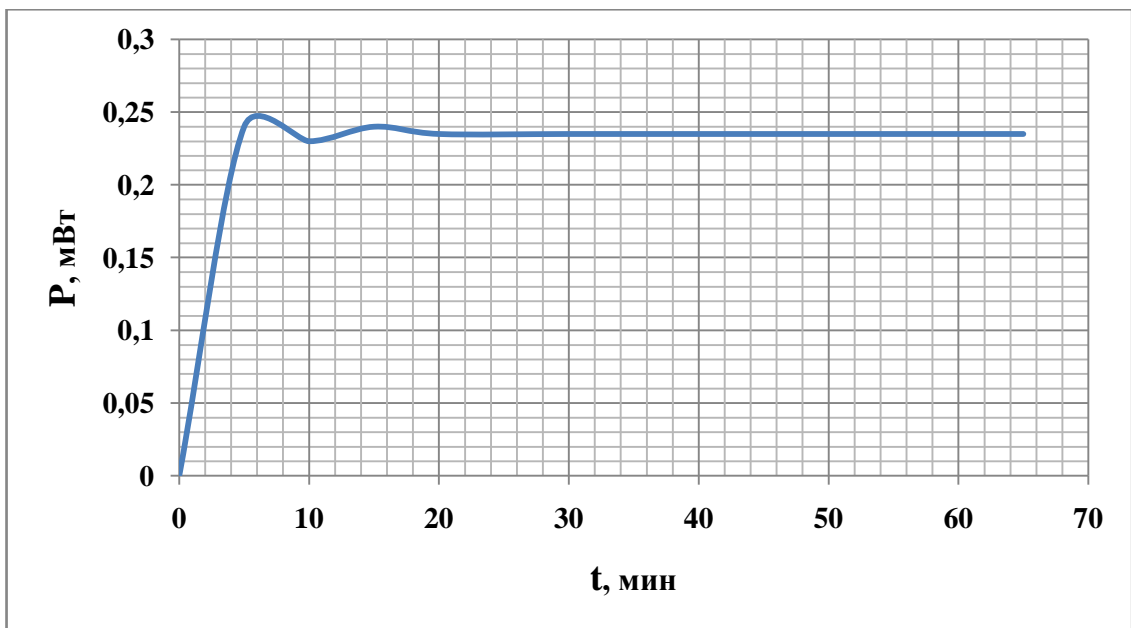


Рис. 2.25. Исследование полупроводникового лазера на стабильность излучения

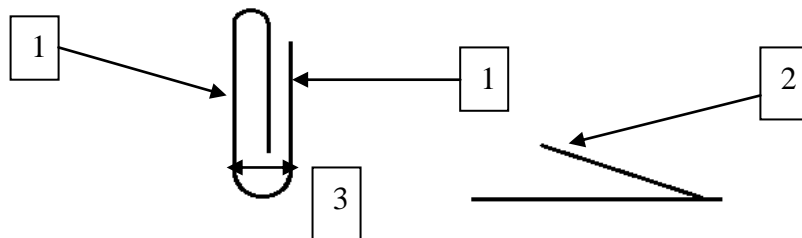


Рис. 2.26. Схематическое изображение чувствительного элемента:

1 – опорные стержни; 2 – давящий стержень; 3 – расстояние между опорными стержнями

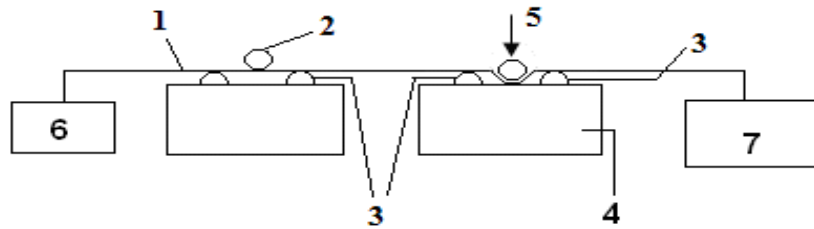


Рис. 2.27. Экспериментальная установка для исследования пригодности различных типов металлических конструкций и определения оптимального расстояния между опорными стержнями: 1 – волоконный световод; 2 – давящий стержень; 3 – опорные стержни; 4 – упругая среда; 5 – воздействие; 6 – источник излучения; 7 – приёмник

В ходе экспериментального исследования на чувствительный элемент производилось одинаковое по степени воздействие и изменялось расстояние между опорными стержнями. Эксперимент проводился три раза для трёх различных диаметров опор. Результаты исследования представлены на рисунках 2.28, 2.29, 2.30.

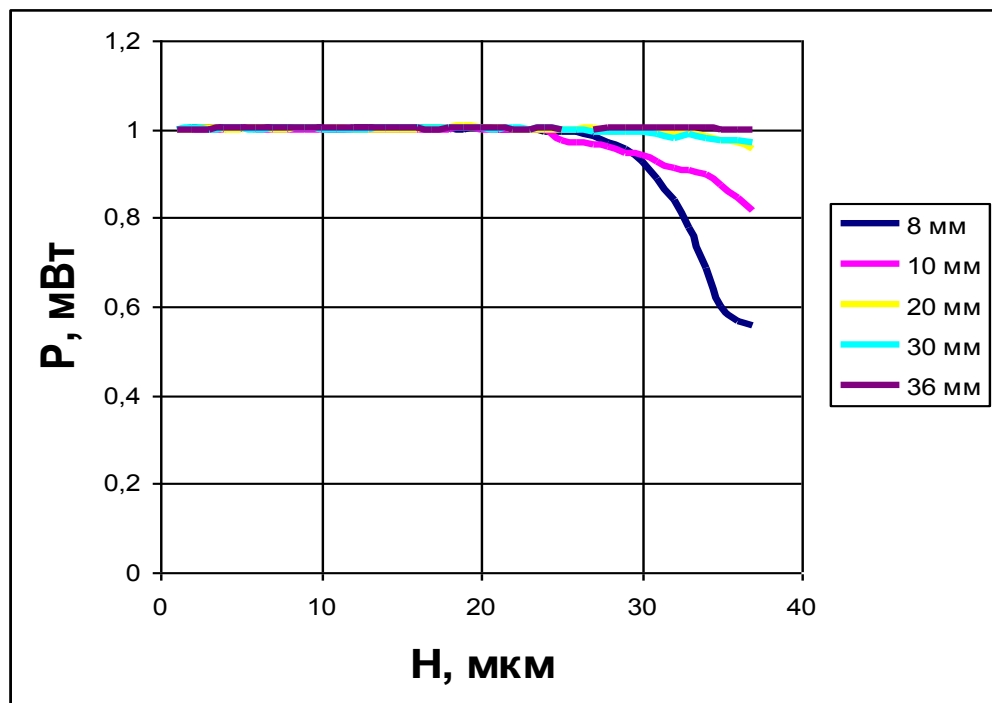


Рис. 2.28. Графики, отображающие зависимость мощности от степени воздействия, при различных расстояниях между опорными стержнями и диаметром стержня  $D=2$  мм, где  $H$  – глубина погружения оптоволоконна в упругую среду



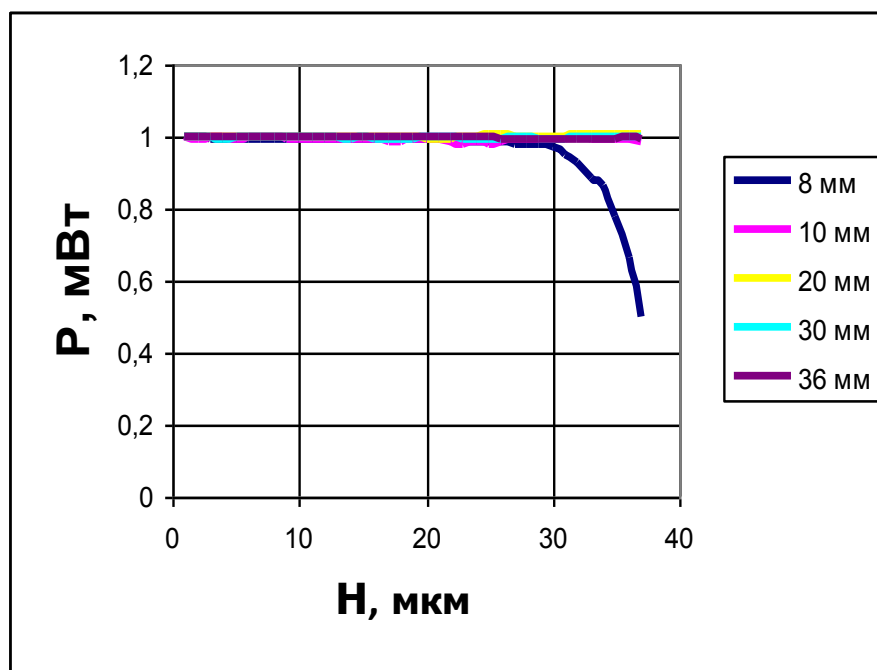


Рис. 2.29. Графики, отображающие зависимость мощности от степени воздействия, при различных расстояниях между опорными стержнями и диаметром стержня  $D = 1,5$  мм, где  $H$  – глубина погружения оптоволокна в упругую среду

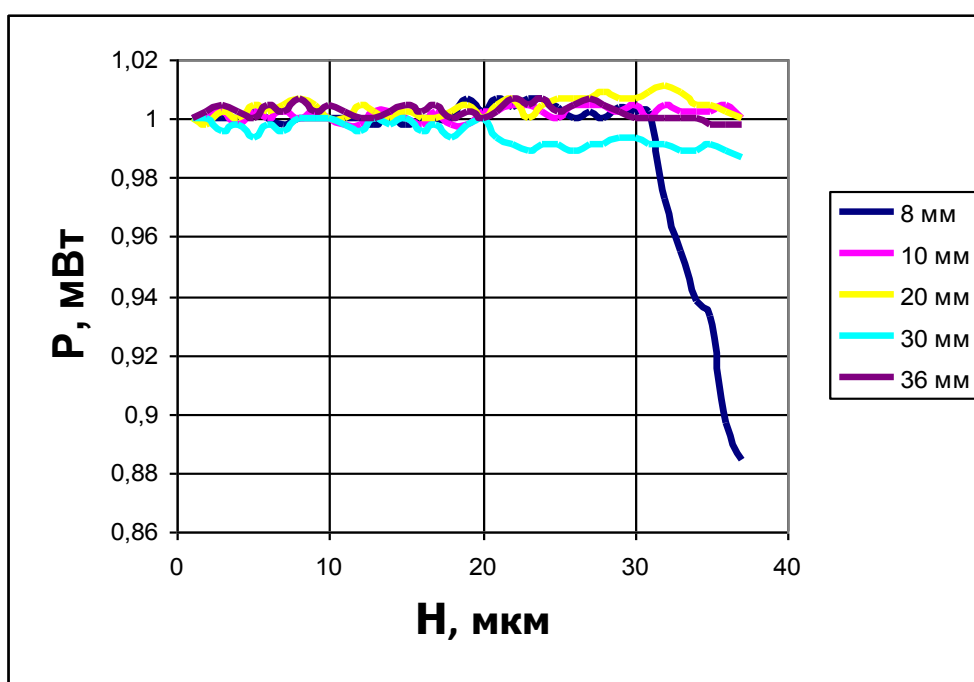


Рис. 2.30. Графики, отображающие зависимость мощности от степени воздействия, при различных расстояниях между опорными стержнями и диаметром стержня  $D = 0,5$  мм, где  $H$  – глубина погружения оптоволокна в упругую среду

Как видно из представленных данных оптимальным расстоянием между опорными стержнями является 8 мм, так как наблюдается наибольшая чувствительность при воздействии. При расстоянии менее 8 мм при воздействии на чувствительный элемент происходит излом оптоволокна. При расстоянии большем, чем 8 мм происходит падение чувствительности элемента. Также важным параметром чувствительного элемента является диаметр стержней. Как видно из представленных результатов, полученных в ходе эксперимента, наиболее подходящими для создания данной конструкции являются металлические стержни с диаметром 2 мм, так как при изгибе ВС наблюдаются максимальные потери мощности оптического излучения. Усреднённые результаты исследований показаны на рисунке 2.31.

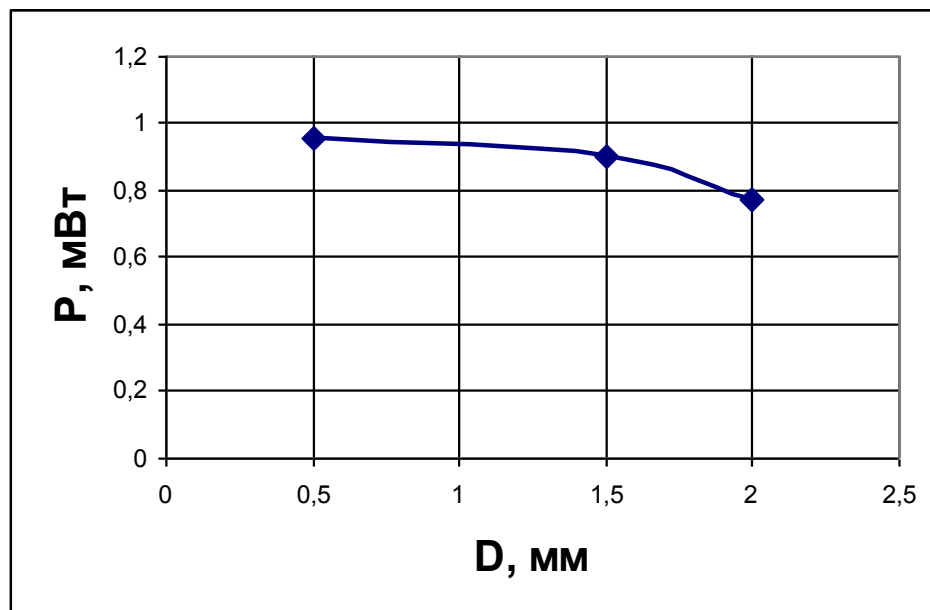


Рис. 2.31. График зависимости мощности на выходе от диаметра металлических стержней

На основании результатов проведенных исследований был создан чувствительный элемент со следующими параметрами: диаметр металлических стержней  $D=2\text{мм}$ , расстояние между опорными стержнями  $L=8\text{мм}$ . Данный чувствительный элемент изображён на рисунке 2.32.

При создании измерительной линии было выбрано оптимальное расстояние между чувствительными элементами такое, чтобы не наблюдалось большой

информативности данных, усложнявших обработку сигнала, и в тоже время отсутствовал существенный риск не зарегистрировать малое воздействие. Расстояния между чувствительными элементами и между измерительными линиями одинаковые. ИЛ объединены после создания в измерительную сеть. Измерительная линия сконструирована на твердой опоре, на которую наклеен упругий материал (упругая среда), в данном случае – поролон [80, 81].

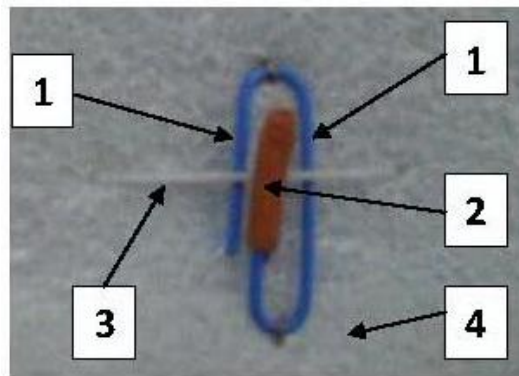


Рис. 2.32. Чувствительный элемент на основе многомодового волоконного световода:  
1 – опорные стержни, 2 – давящий стержень, 3 – волоконный световод, 4 – упругая среда

Наличие упругой среды необходимо, чтобы:

- после воздействия волоконный световод принимал исходное положение
- не произошло чрезмерного прогиба волокна, который может привести к излому, а, следовательно, и к выходу из строя ЧЭ.
- компенсировались изменения длины чувствительного участка, которое может привести к чрезмерному натяжению ВС и в конечном итоге к обрыву волокна.

Схема измерительной линии представлена на рисунке 2.33. В ЧЭ применялось стандартное многомодовое волокно с буферным покрытием, со ступенчатым показателем профиля преломления, диаметр сердцевины 50мкм, диаметр оболочки 125мкм  $NA=0,215$ . Таким образом, для макета измерительной сети была создана конструкция ЧЭ и ИЛ, изображённая на рисунке 2.34.

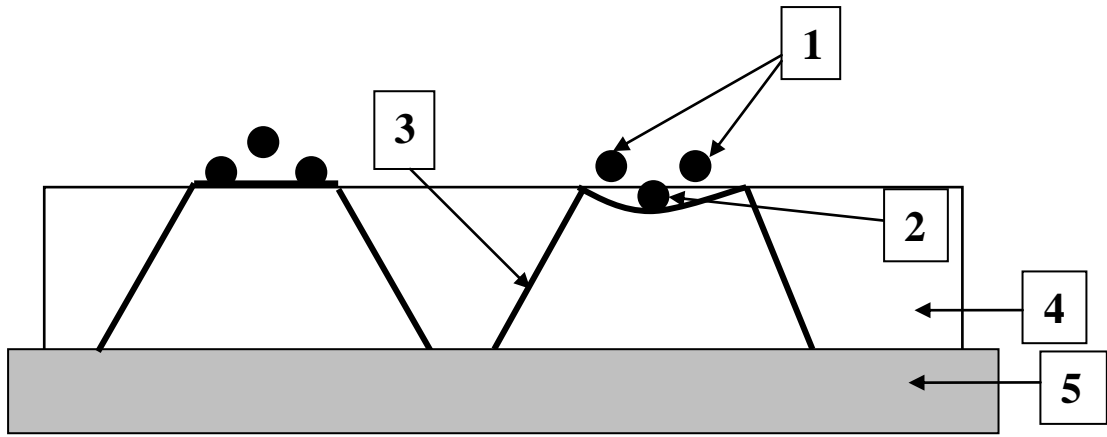


Рис. 2.33. Схема измерительной линии в разрезе.

Слева – исходное состояние, справа – деформация ВС. 1- опорные стержни, 2-вдавливающий стержень, 3 – волоконный световод, 4 – упругая среда, 5 – твердая опора

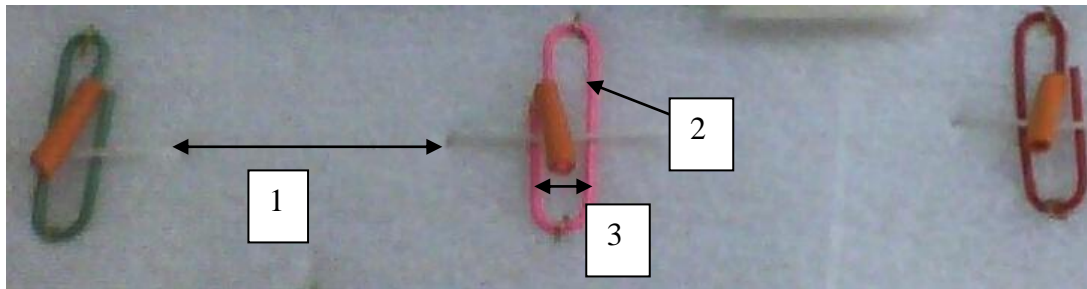


Рис. 2.34. Внешний вид измерительной линии:

1 – расстояние между чувствительными элементами; 2 – опорные стержни, с диаметром 2 мм;  
3 – расстояние между опорными стержнями, равное 8 мм

#### 2.4.2 Принципы создания чувствительного элемента на основе одномодового волоконного световода

Для создания чувствительного элемента была выбрана конструкция аналогичная представленной на рисунке 2.26. При исследовании пригодности различных типов одномодовых волоконных световодов использовалась установка подобная представленной на рисунке 2.27. При разработке чувствительного элемента были предложены 2 типа ВС: FutureGuide-SR155 и Dummy Optical Fiber.

В ходе экспериментального исследования на чувствительный элемент производилось одинаковое по силе воздействие для этих типов ВС. Результаты исследований представлены на рисунке 2.35(*а*, *б*).

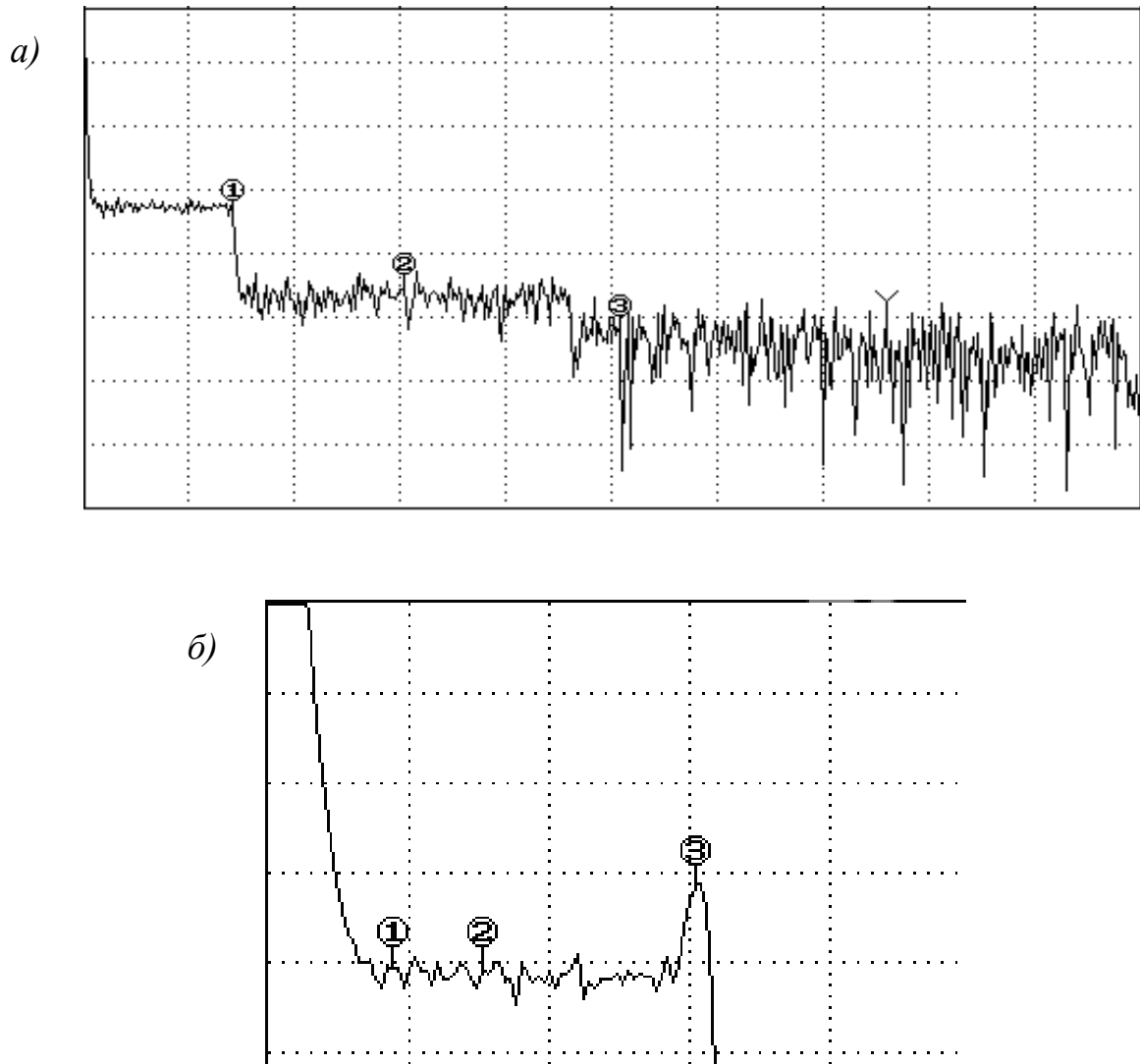


Рис. 2.35. *а*) Рефлектограмма воздействия на ЧЭ с ВС FutureGuide-SR155;

*б*) Рефлектограмма воздействия на ЧЭ с ВС Dummy Optical Fiber

Как видно из представленных результатов второй тип волоконного световода не подходит для использования в разрабатываемой конструкции, так как при изгибе ВС потери мощности оптического излучения за счет вытекающих мод становятся существенными, а так же возникает сильное френелевское отражение. Вследствие данных факторов происходит потеря информации от ВС расположенного после изгиба [93].

В разделе 2.2.2. было показано, что мощность отраженного оптического излучения при изгибе ВС зависит от радиуса кривизны волоконного световода, что в свою очередь зависит от расстояния между опорными стержнями (рис. 2.26). Для выбора оптимального расстояния между ними было проведено исследование, в котором воздействие на чувствительный элемент производилось двумя, существенно, отличными массами ( $m_1=5m_2$ ,  $m_2=1,5\text{кг}$ ), а так же изменялось расстояние между стержнями. Результаты исследования представлены на рисунках 2.36(*а*, *б*), 2.37(*а*, *б*) и 2.38(*а*, *б*).

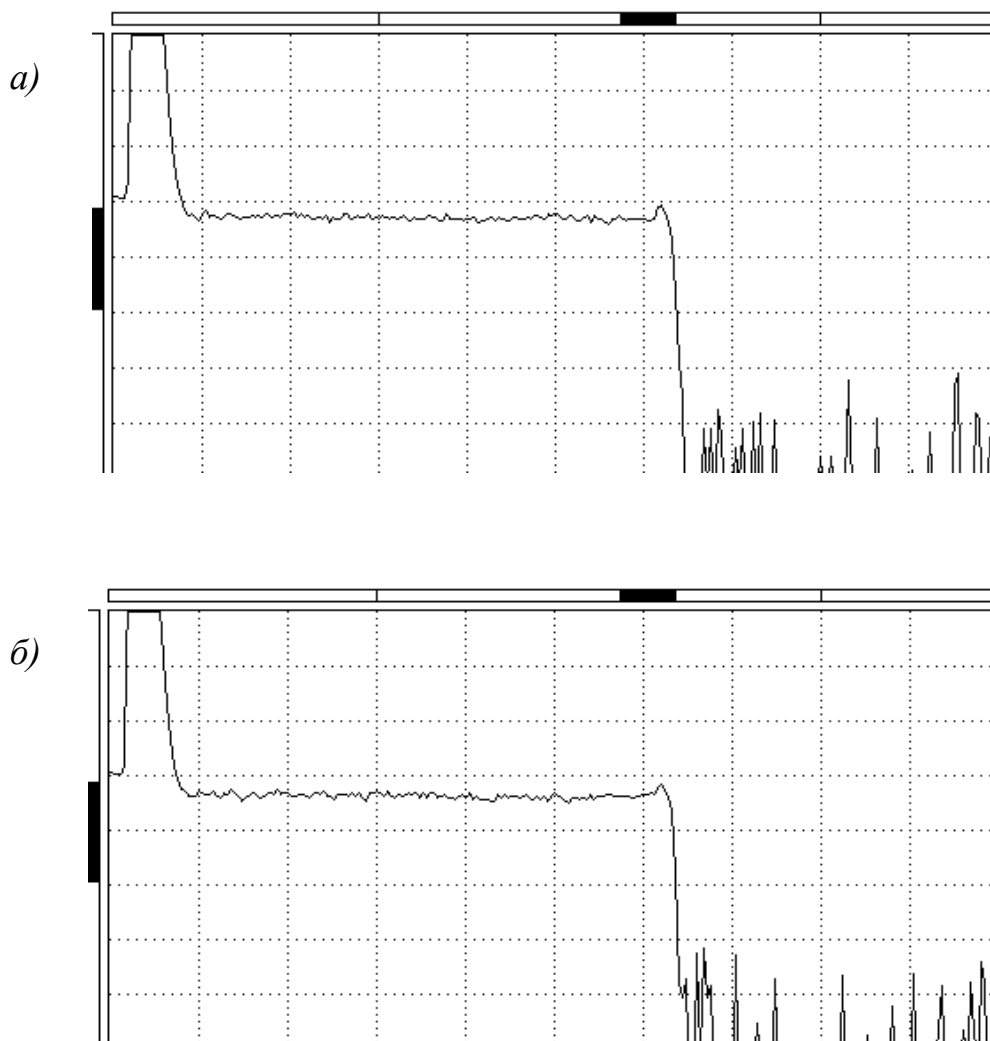


Рис. 2.36. *а*) Рефлектограмма при воздействии массы  $m_1$  и расстоянии 10 мм;  
*б*) Рефлектограмма при воздействии массы  $m_2$  и расстоянии 10 мм

Как видно из представленных данных оптимальным расстоянием между опорными стержнями является 15мм (рис. 2.38(*a, б*)), так как при меньшем расстоянии при воздействии на чувствительный элемент происходит потеря данных от ЧЭ расположенных дальше на измерительной линии (рис. 2.36(*a, б*)). При расстоянии больше чем 15мм присутствует существенный риск не зарегистрировать малое воздействие (рис. 2.37(*a, б*)).

На основании результатов проведенных исследований был создан чувствительный элемент и измерительная линия.

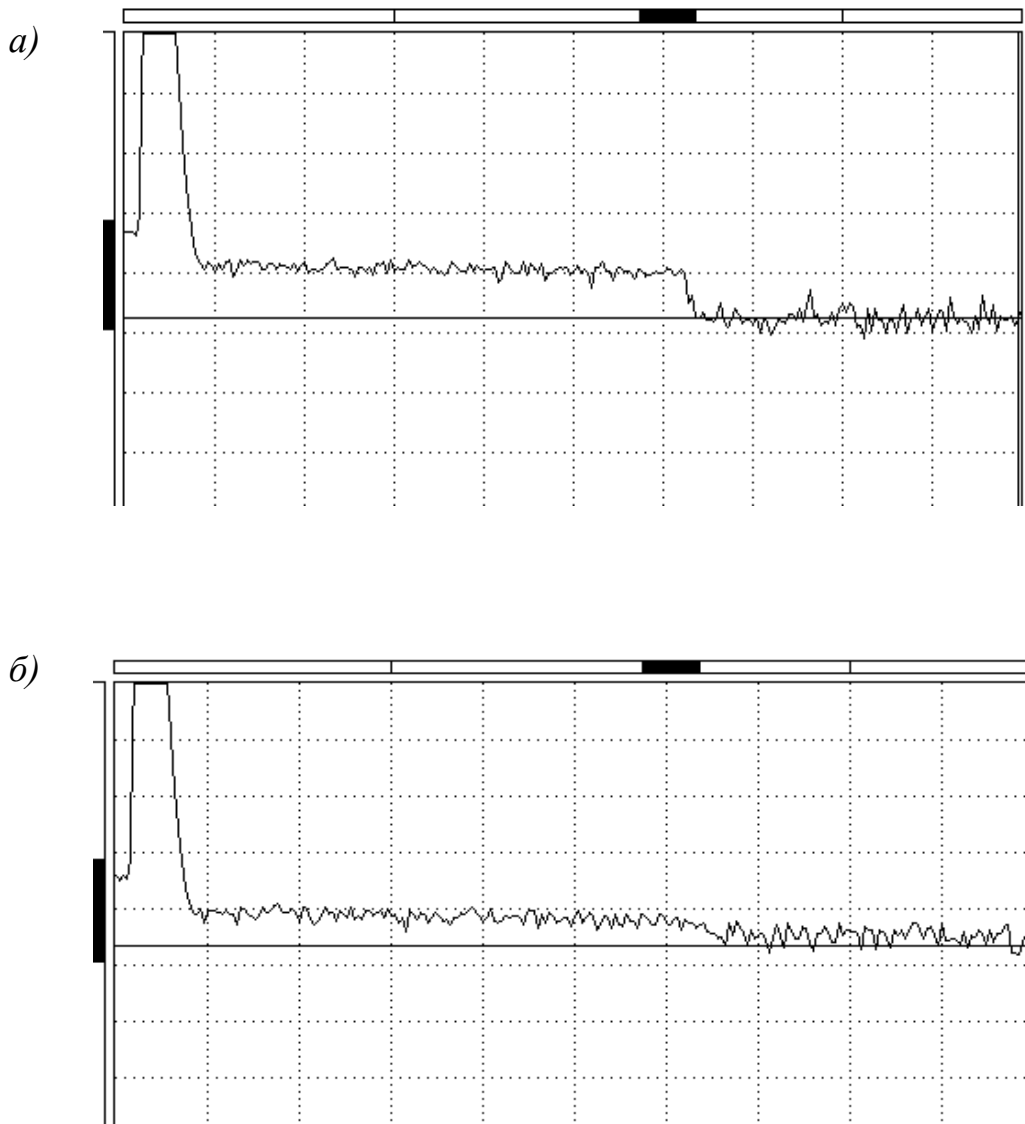


Рис. 2.37. *a)* Рефлектограмма при воздействии массы  $m_1$  и расстоянии 19 мм;  
*б)* Рефлектограмма при воздействии массы  $m_2$  и расстоянии 19 мм

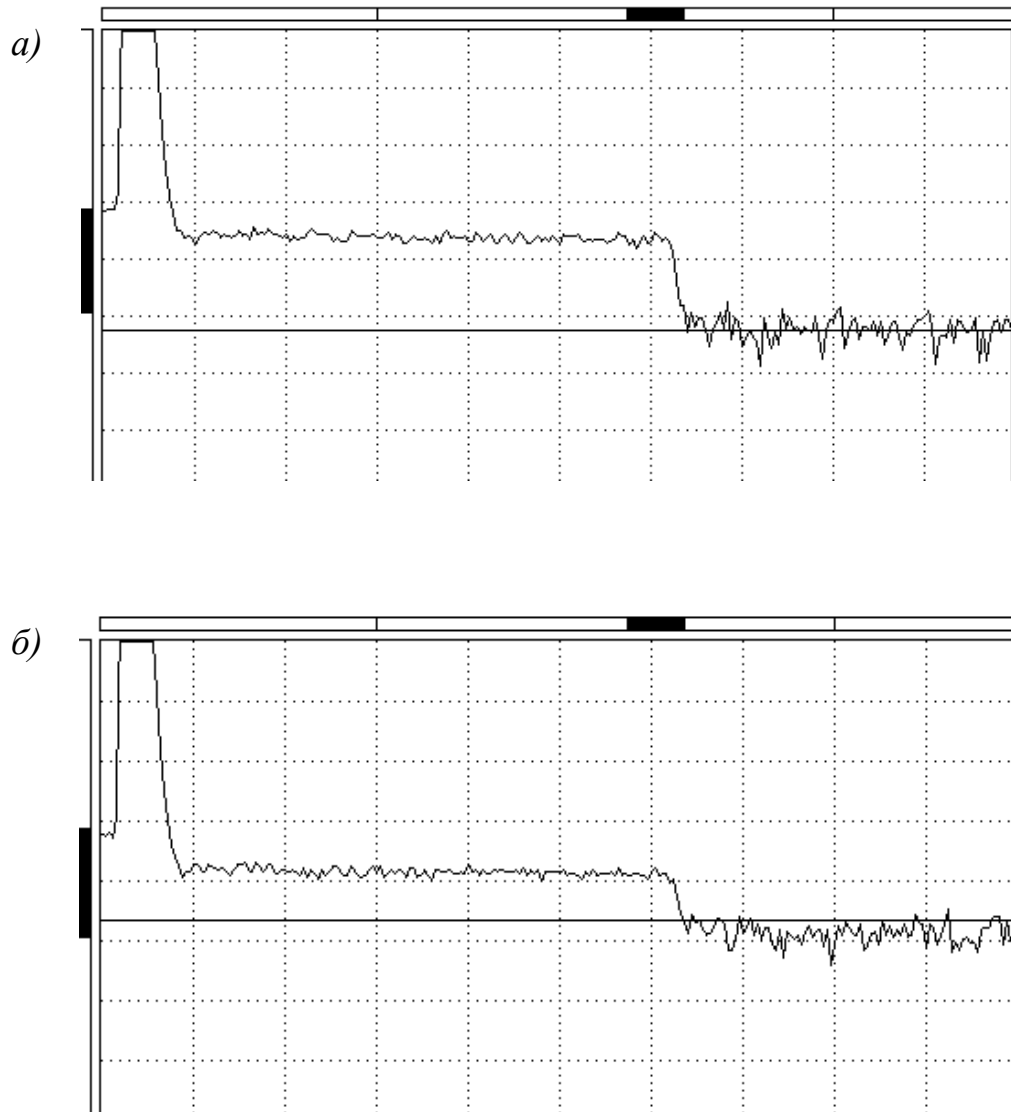


Рис. 2.38. а) Рефлектограмма при воздействии массы  $m_1$  и расстоянии 15 мм;  
 б) Рефлектограмма при воздействии массы  $m_2$  и расстоянии 15 мм

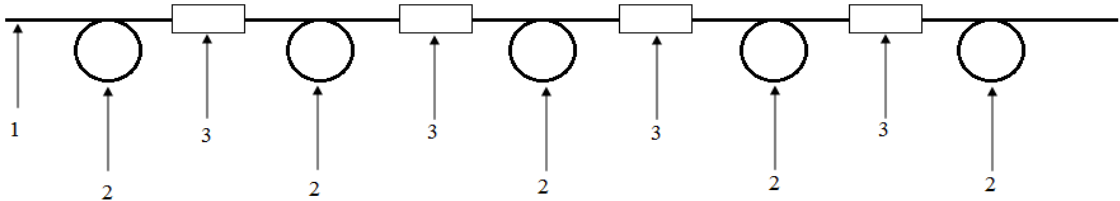
### 2.4.3 Создание волоконно-оптической измерительной линии на основе одномодового волоконного световода

Схематическое изображение измерительной линии на основе одномодового волоконного световода представлено на рисунке 2.39.

Для ввода оптического излучения в измерительную линию необходимо наличие оптического коннектора на одном из концов данной линии. Добиться этого можно двумя путями:



1. Созданием оптического коннектора непосредственно на конце волоконного световода – оконцовка.
2. Сращиванием конца измерительной линии с отрезком кабеля, на котором с одной стороны коннектор уже был смонтирован в заводских условиях.



1 – одномодовый волоконный световод; 2 – кассета с запасом ВС; 3 – ЧЭ

Рис. 2.39. Измерительная линия на основе одномодового волоконного световода

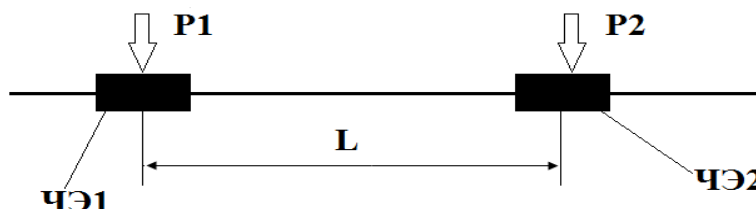
В данной работе был выбран второй вариант, так как процесс оконцовки достаточно сложен и трудоемок. Сращивание можно выполнять для кабелей различного типа (одномодовых или многомодовых) двумя способами — сваркой или монтажом неразъемных соединителей (механических сплайсов). В обоих случаях нужен определенный набор инструментов, приспособлений и расходных материалов. При реализации использовался первый способ, так как он дает минимальные потери в месте соединения. Второй же способ не подходит по причине того, что после прохождения сродки слишком велико затухание оптического излучения.

Перед непосредственно самой сваркой оптических волокон необходимо произвести ряд действий:

- На одно из волокон надеваются специальные гильзы – КДЗС (комплект для защиты стыка), состоящие из двух термоусадочных трубок и силового стержня.
- С концов волокон (2 – 3см) снимается цветной лак и защитный слой, волокна протираются спиртом.

- Зачищенное волокно скалывается прецизионным скалывателем. Плоскость скола волокон должна быть перпендикулярна оси волокна. Допустимое отклонение — до  $1,5^\circ$  на каждый скол.
- Волокна, предназначенные для сварки, укладываются в зажимы сварочного аппарата (V-образные канавки).
- Под микроскопом с помощью манипуляторов происходит их совмещение (юстировка). В современных сварочных аппаратах юстировка происходит автоматически.

Способ и качество юстировки – наиболее принципиальный момент. Выравнивание волокон может выполняться по оболочке с их центрированием в V-образном пазу, а также по сердцевине: по профилю преломления волокна или максимизацией передаваемого через выравниваемые волокна сигнала. Сварка волокон выполнялась путем разогрева концов волокон в электрической дуге и их соединения. Далее аппарат осуществляет проверку прочности соединения посредством механической деформации и оценивает затухание, вносимое стыком. На конечном этапе КДЗС сдвигается на место для сварки. Этот участок помещается в тепловую камеру, где происходит термоусадка КДЗС. Для более надежной защиты сваренные волокна укладываются в сплайс-пластину, кассету оптической муфты или кросса. Для обеспечения оптимальной работы разработанной измерительной линии необходимо знать минимальное расстояние между чувствительными элементами. Для исследования использовалась установка, представленная на рисунке 2.40.



$L$  – расстояние между чувствительными элементами,  $P1$  и  $P2$  – воздействия, производимые, соответственно, на ЧЭ1 и ЧЭ2

Рис. 2.40. Установка для исследования минимального расстояния между чувствительными элементами

В ходе проведения эксперимента на чувствительные элементы производилось одновременное одинаковое воздействие на разных расстояниях (рис. 2.41, 2.42, 2.43). Было установлено, что при расстоянии между ЧЭ менее 3 метров, на рефлектограмме невозможно различить эти воздействия (рис. 2.41).

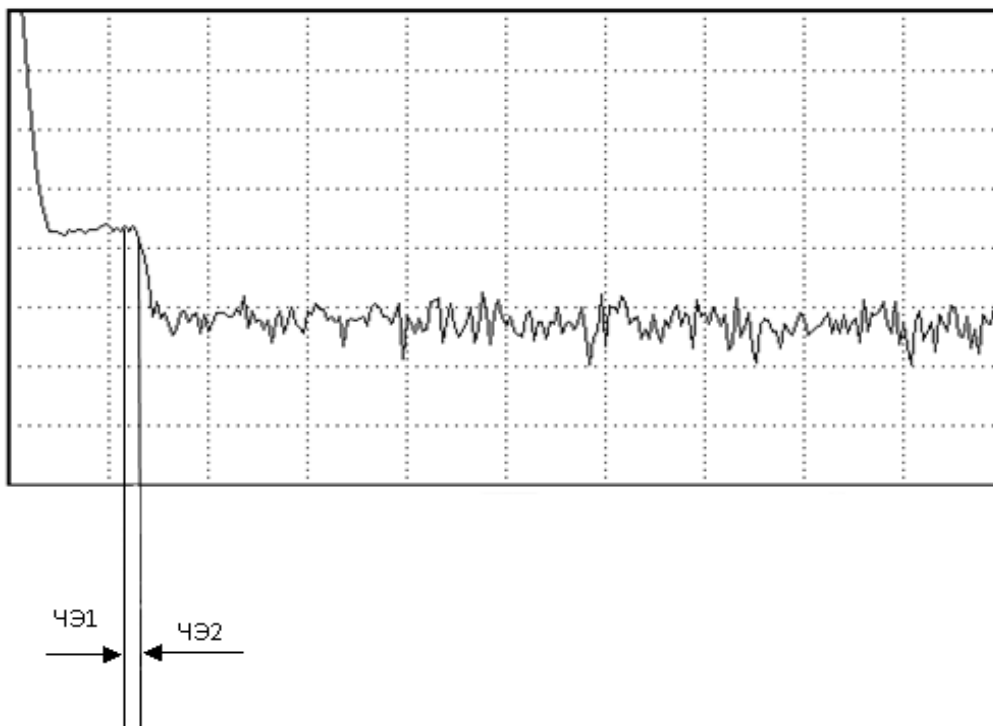


Рис. 2.41. Рефлектограмма измерительной линии при расположении ЧЭ на расстоянии  
~3м

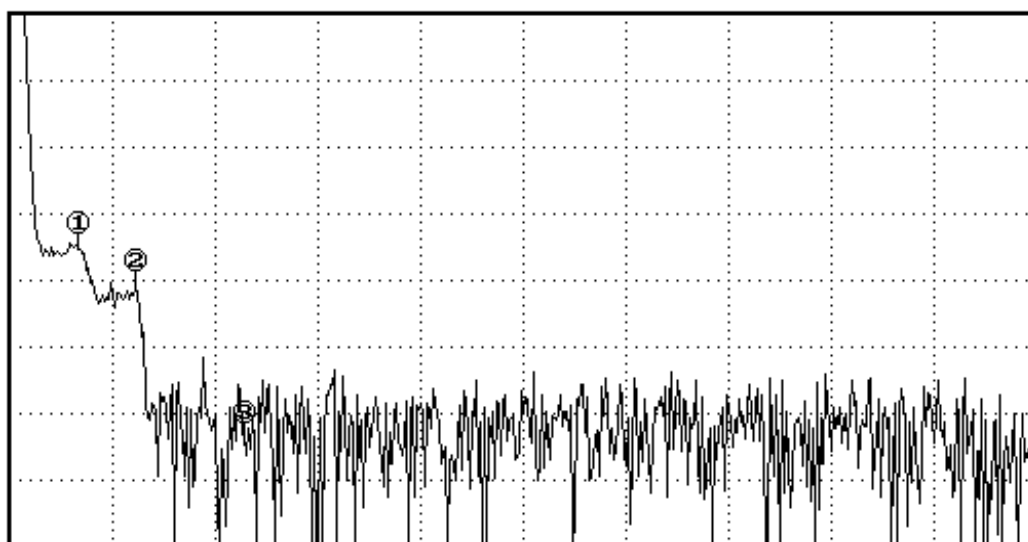


Рис. 2.42. Рефлектограмма измерительной линии при расположении ЧЭ на расстоянии  
~8м

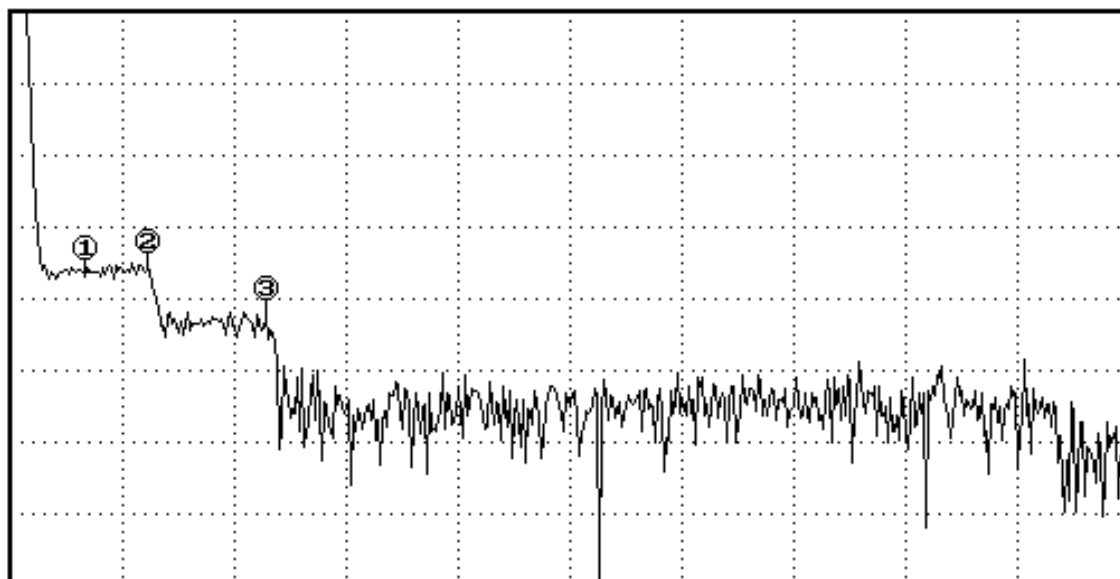


Рис. 2.43. Рефлектограмма измерительной линии при расположении ЧЭ на расстоянии ~19м

При увеличении расстояния до 8 метров можно четко разделять пики отражения, возникающие при изгибе волоконного световода (рис. 2.42), однако для исключения взаимного воздействия помех, возникающих при одновременном воздействии на оба участка измерительной линии, было выбрано оптимальное расстояние между ЧЭ ~19 метров (рис. 2.43).

Так как разработанная измерительная линия предназначена для локализации места воздействия, то ее важным параметром является разрешающая способность, т.е. количество одновременных различных воздействий на данную измерительную линию. Для экспериментального исследования на рефлектограмме были размещены маркеры обозначающие местоположение чувствительных элементов. Для точного поиска места положения ЧЭ на рефлектограмме на каждый чувствительный элемент последовательно было произведено максимально допустимое воздействие (рис. 2.44). После установления маркеров было произведено одновременное воздействие различными массами на чувствительные элементы под маркерами 2, 3 и Y (рис. 2.45).

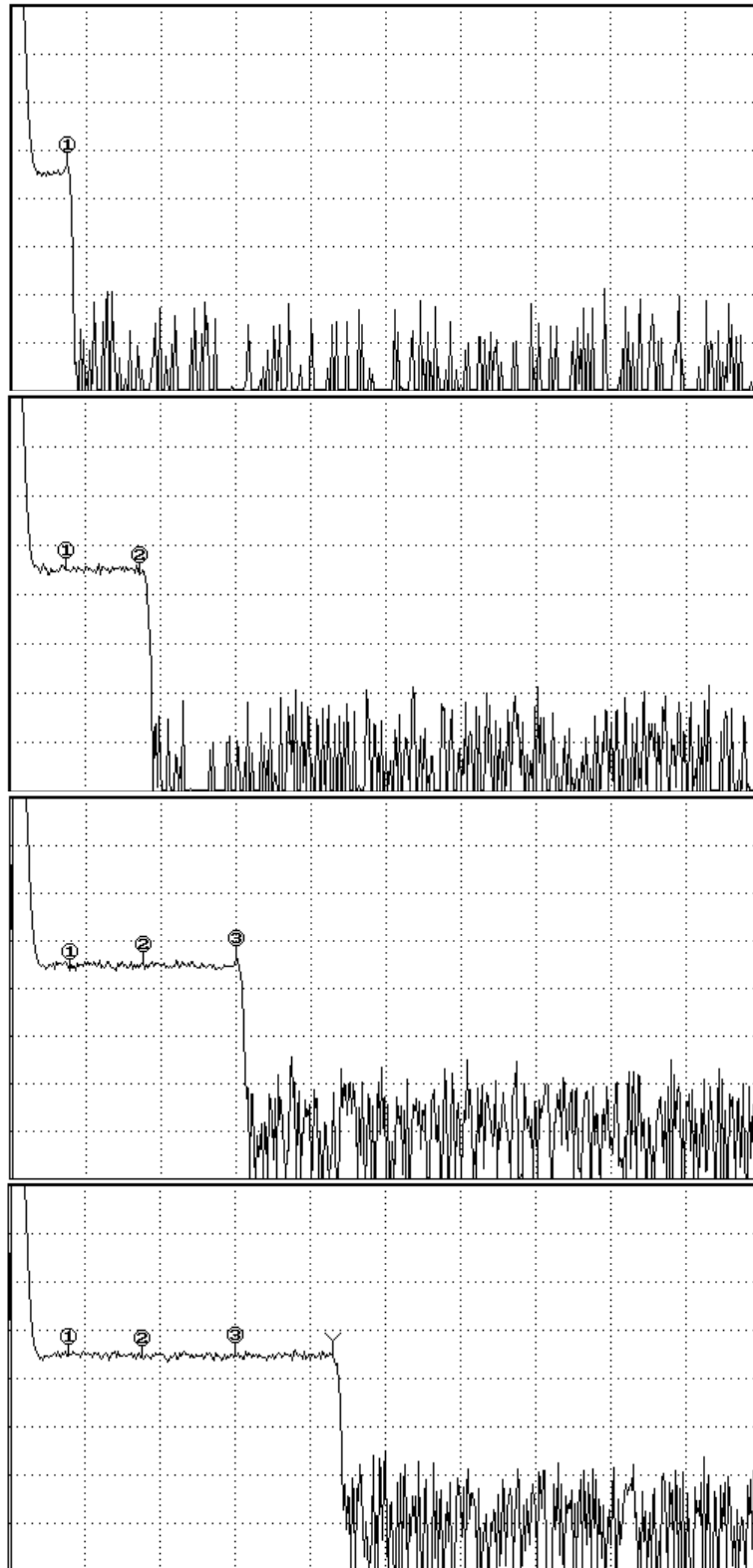


Рис. 2.44. Рефлектограммы измерительной линии в процессе поиска места положения ЧЭ

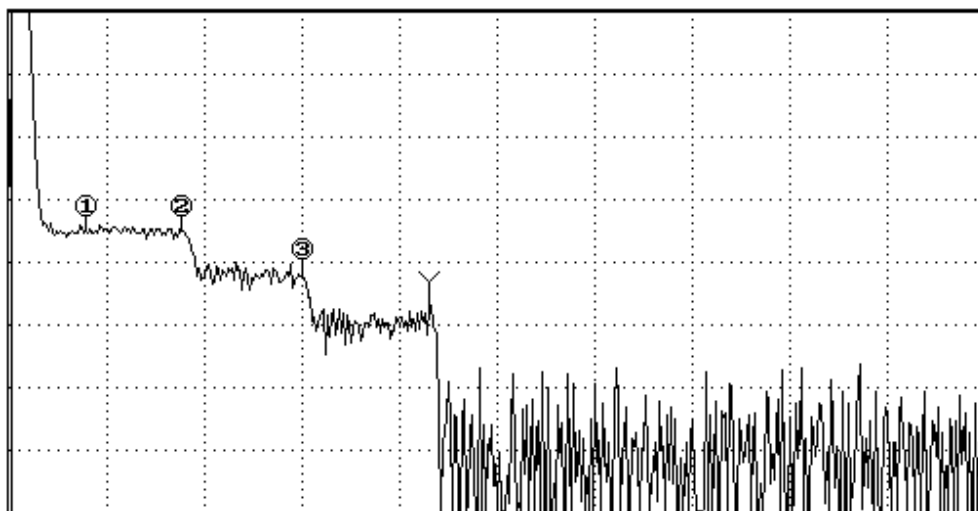


Рис. 2.45. Рефлектограмма при одновременном воздействии на измерительную линию

Проанализировав полученные данные (разделы 2.4.2, 2.4.3), а так же, используя, результаты исследования измерительных линий на основе многомодовых волоконных световодов (раздел 2.4.1) был создан макет волоконно-оптической измерительной сети.

#### **2.4.4 Макет волоконно-оптической измерительной сети системы охраны периметра**

Для создания макета ВОИС охраны периметра использовались созданные измерительные линии на основе одномодовых и многомодовых ВС [80, 81]. Макет представляет собой площадку размером 0,64×0,80 м, условно разбитую на 20 сегментов (рис. 2.46а). Амплитудные ВОД на основе многомодовых ВС (по 10 ЧЭ в ИЛ) уложены параллельно так, что через каждый сегмент проходят две линии. При этом на каждой ИЛ в сегменте располагается по два ЧЭ одного датчика, то есть четыре ЧЭ на сегмент. При создании измерительной линии на основе одномодового ВС использовалась конструкция ЧЭ аналогичная той, что и для ЧЭ на основе многомодового ВС (рис. 2.46б). Для измерительной линии на основе одномодового ВС применяли волокно типа FutureGuide-SR155, так как при изгибе ВС данного типа потери мощности оптического излучения за счет вытекающих мод становятся менее существенными и наблюдается наименьшее френелевское отражение. Для данной измерительной линии было выбрано

оптимальное расстояние между чувствительными элементами равно 19 метрам. Для того чтобы расстояние между ЧЭ в измерительной сети было меньше, участки ВС между ними наматывались на катушки, находящиеся на нижнем уровне измерительной сети. Учитывая данные характеристики, была создана измерительная линия на основе одномодового ВС в виде OTDR-датчика.

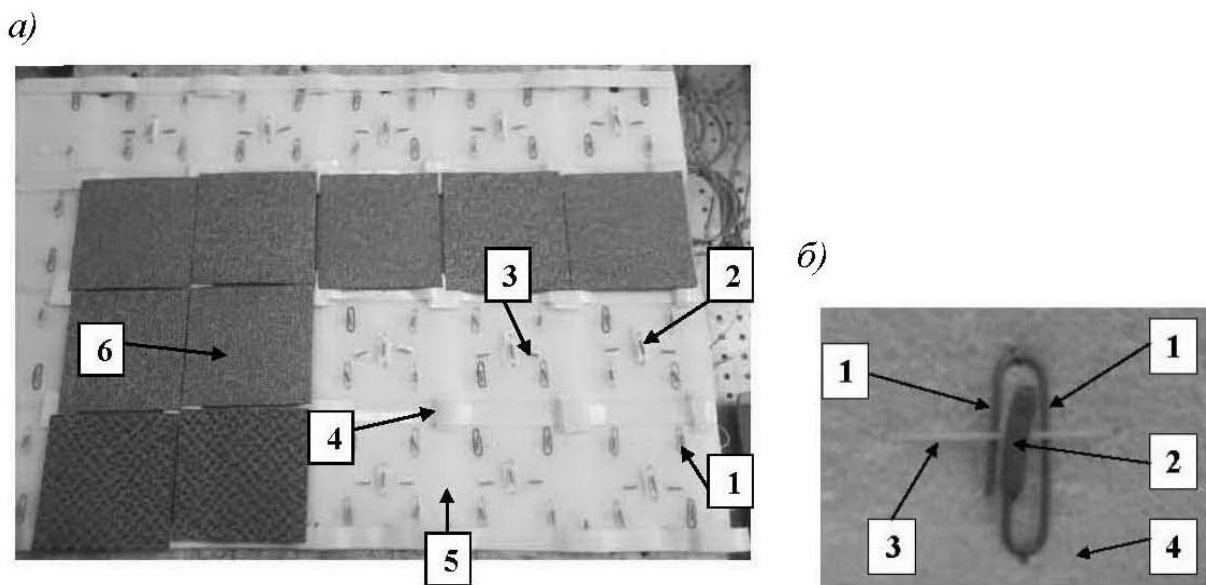


Рис. 2.46. а) Макет волоконно-оптической сети охраны периметра: 1 – чувствительный элемент на основе многомодового волоконного световода (ВС), 2 – чувствительный элемент на основе одномодового ВС, 3 – направляющая для одномодового ВС, 4 – пружинящая подкладка ВС, 5 – упругая подложка, 6 – защитное покрытие из твердого материала; б) Чувствительный элемент на основе многомодового волоконного световода: 1 – опорные стержни, 2 – давящий стержень, 3 – волоконный световод, 4 – упругая среда

Для мониторинга измерительной линии на основе одномодового ВС применяется оптический рефлектометр, а для измерительных линий на основе многомодового ВС была разработана система обработки выходного сигнала, состоящая:

- Блока фотоприемников;
- АЦП;
- Модуля обработки данных (комплекс специализированных программ обработки данных с ИЛ).

Изменение мощности в многомодовых ВС регистрируется блоком фотоприемников, через АЦП связанных с компьютером. На компьютере установлено программное обеспечение для обработки данных. Фотоприёмники изготавливались по схеме, изображённой на рисунке 2.47.

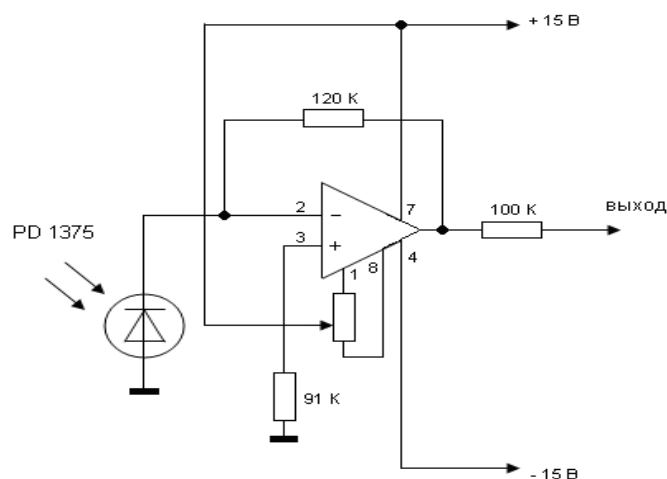


Рис. 2.47. Схема изготовления фотоприёмника:

1, 8 – балансировка, 2- вход инвентирующий, 3 – вход неинвентирующий, 4 – напряжение питания ( $-U_{п1}$ ), 7 – напряжение питания ( $U_{п2}$ )

Также в процессе изготовления фотоприёмника использовался фотоприёмный модуль PD-1375, изготовленный на основе фотодиода, состыкованного с многомодовым ВС. Таким образом, был изготовлен блок фотоприёмников (рис. 2.48), состоящий из шести единиц (по одному фотоприёмнику на каждую измерительную линию).

Впоследствии созданные измерительные линии объединялись в сеть, поэтому возникла необходимость использования многоканального разветвителя выходного сигнала  $1 \times 8$ . Используемое многоканальное устройство изображено на рисунке 2.49. В данное устройство вмонтирован древовидный разветвитель («сплиттер»), служащий для распределения потоков оптической мощности. В созданной установке сплиттер является одноволоконным симметричным, так как необходимо, чтобы мощность распределялась одинаково между всеми



выходными портами. Чтобы устранить «мертвые» зоны между чувствительными элементами было использовано покрытие из жесткого материала, так же выполняющие защитную функцию. Данное покрытие было выполнено в виде разделенных квадратных сегментов для обеспечения оперативного доступа к измерительной линии в случае поломки.

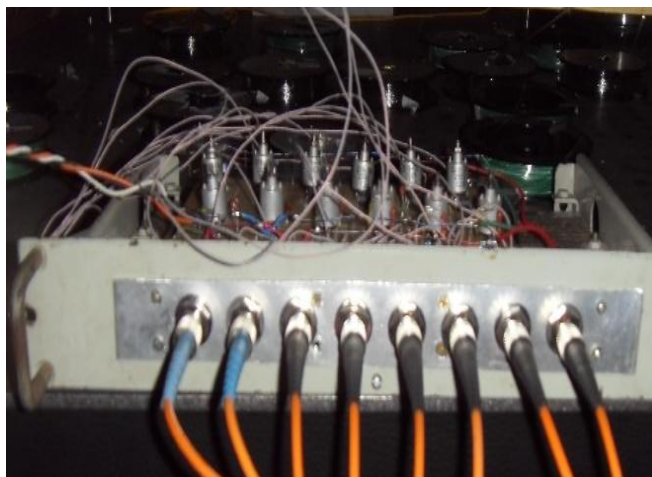


Рис. 2.48. Блок фотоприёмников



Рис. 2.49. Многоканальный разветвитель выходного сигнала 1×8

Таким образом, был реализован макет сети системы охраны периметра, изображённый на рисунке 2.50. [80, 81]. Фактически представлена ВОИС, которая может функционировать как независимая измерительная сеть, так и служить измерительным сегментом РВОИС больших размеров. Подходы и алгоритмы обработки данных с ИЛ, предлагаемые в следующей 3 главе для данной системы,

в частности, также можно использовать и для обработки данных РВОИС сегментарного типа в целом [80, 81].



Рис. 2.50. Общий вид макета ВОИС системы охраны периметра

## 2.5 ВЫВОДЫ

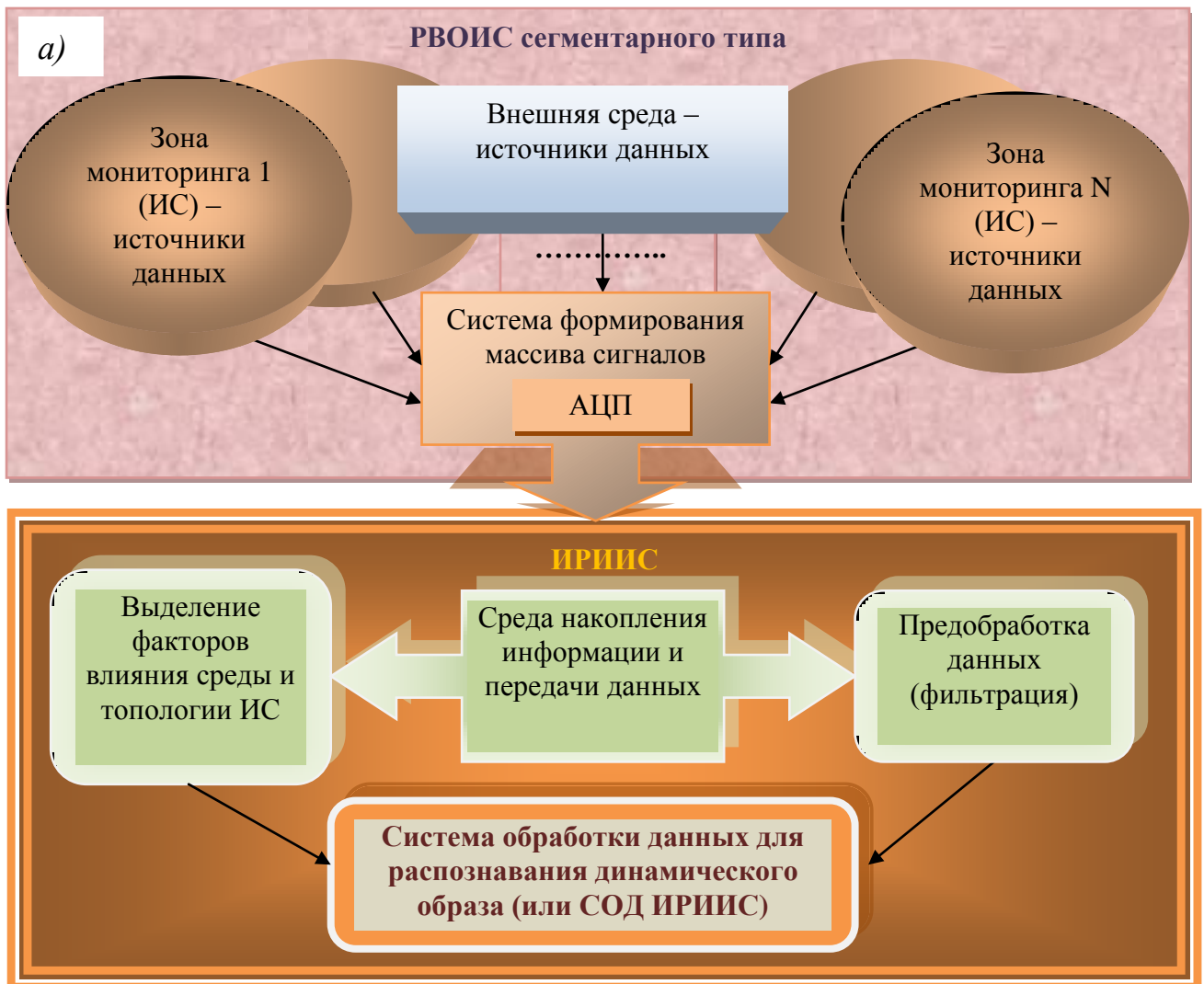
Идея создания ИРИИС на основе РВОИС сегментарного типа происходит из целого ряда преимуществ, которые имеет РВОИС перед измерительными системами других типов. Карта укладки измерительных линий такой сети должна быть оптимизирована под задачи, для решения которых она создаётся. Оптимизацию можно осуществить по критерию качества укладки, определив ранг матрицы измерительной системы. Системы интегрального типа, а РВОИС относится к ним, применимы в случае необходимости сохранения высокой разрешающей способности.

Физическая реализация макета ВОИС системы охраны периметра подтверждает целесообразность системного подхода и комбинированного принципа построения РВОИС. Сочетание разного типа измерительных линий с амплитудными ВОД на основе многомодовых ВС и с OTDR-датчиками на основе одномодового ВС, применённое в разработке макета, позволяет недостатки одних типов линий компенсировать преимуществами других. Обработка сигналов РВОИС может происходить параллельно с применением известных и

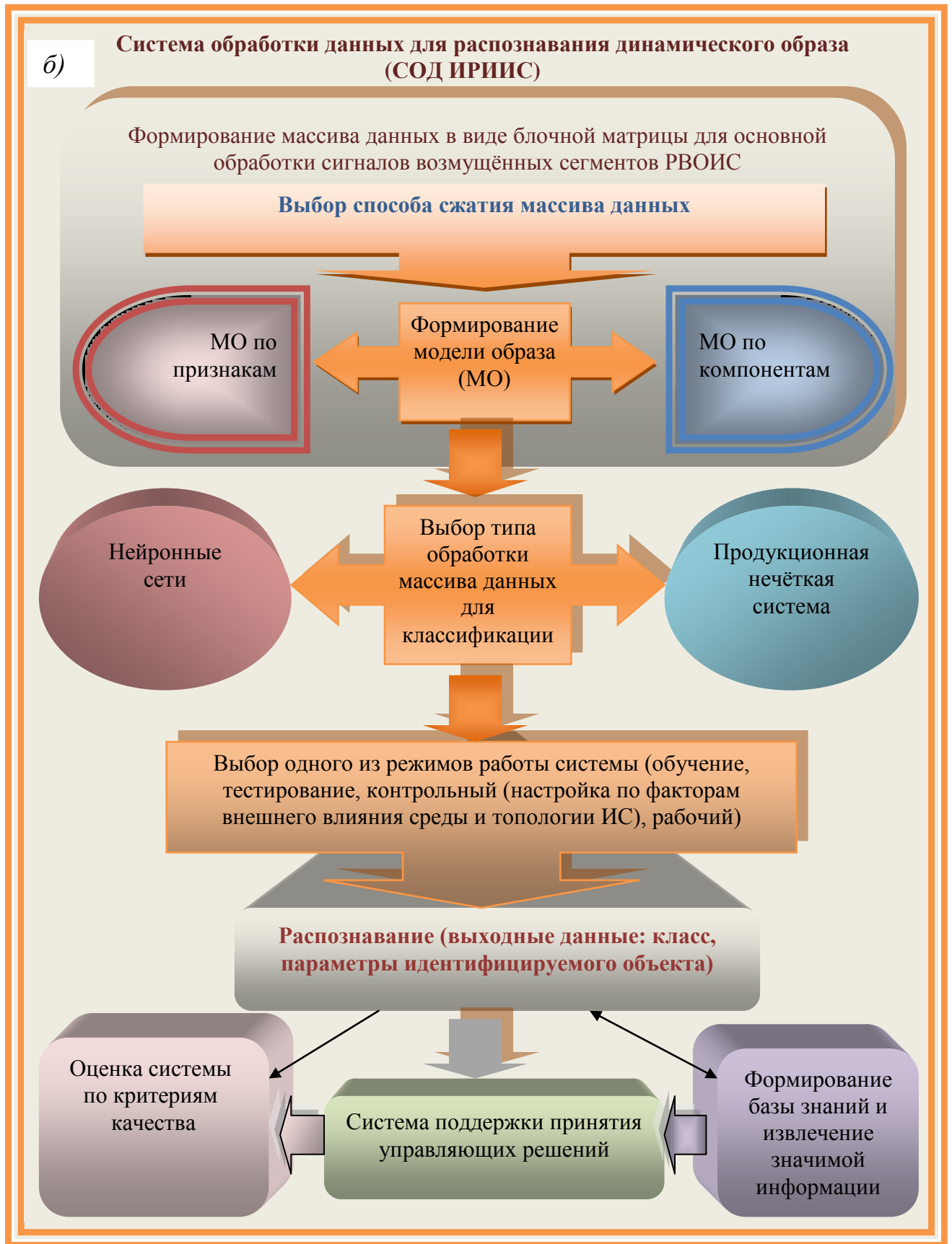
предложенных в главе методов, аккумулируясь в единое решение поставленной в работе задачи.

### 3. СИСТЕМА ОБРАБОТКИ ДАННЫХ РАСПРЕДЕЛЁННОЙ ВОЛОКОННО-ОПТИЧЕСКОЙ СЕТИ СЕГМЕНТАРНОГО ТИПА

Для решения задачи построения ИРИИС на основе РВОИС сегментарного типа была предложена схема (рис. 1.1.). Общая структурно-функциональная модель реализуемой системы показана на рисунке 3.1(*а, б*), в ней процесс распознавания динамических образов представлен в виде последовательности взаимосвязанных функций: от сбора данных об объекте (блок I) до обработки (блок II) [1, 2, 153].



Структурно-функциональная модель ИРИИС на основе РВОИС сегментарного типа (ИС – измерительный сегмент сети, АЦП – аналого-цифровой преобразователь)



Система обработки данных для распознавания динамического образа в структурно-функциональной модели ИРИИС на основе РВОИС сегментарного типа

Рис. 3.1.

При подготовке, на уровне построения концептуальной модели разрабатываемой ИРИИС на основе РВОИС сегментарного типа, учитывались все признаки интеллектуальности (развитые коммуникативные способности, умение решать сложные плохо формализуемые задачи, способность к обучению [154]) с целью развить их в достаточной степени в предложенной модели системы.

Для извлечения значимой информации и формирования модели образа МО из данных РВОИС используют многомерные статистические методы (метод главных компонент, снижение размерности признакового пространства, корреляционный анализ количественных переменных) в комплексе с индуктивными методами построения деревьев решений или с нейронными сетями. Формирование МО осуществлялось в результате применения созданных программных модулей в виде генератора сенсорных данных РИС, позволяющего гибко задавать и учитывать значимые факторы влияния среды, и модуля преобразования данных, полученных от макета ВОИС охраны периметра, что значительно расширило возможности исследований и анализа.

Экспериментальные исследования ИРИИС показали необходимость интеграции интеллектуальных и традиционных информационных технологий, комбинированного использования различных методов представления и вывода знаний, усложнения архитектуры информационных систем. Таким образом, возможность выбора и сочетания различных подходов, способов и методов в ИРИИС повышает эффективность функционирования системы, оценки которой по критериям качества можно разделить на две основные группы: точности и пригодности системы к выполнению поставленных задач.

## **3.1 СИСТЕМА ОБРАБОТКИ ДАННЫХ ДЛЯ РАСПОЗНАВАНИЯ ДИНАМИЧЕСКОГО ОБРАЗА С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ**

### **3.1.1 Разработка генератора выборки примеров сенсорных данных**

В рамках исследования различных подходов и методов для создания алгоритмического комплекса и специализированного программного обеспечения обработки данных РВОИС был разработан генератор сенсорных данных РИС.

Генератор сенсорных данных представляет собой программу, которая сначала отдельно конструирует возможные траектории движения идентифицируемых объектов (с помощью "Конструктора траекторий движения") и сами объекты (с помощью "Конструктора объектов"), а затем, с помощью "Конструктора сценариев", указывают какие объекты с какими траекториями необходимо использовать при генерации выборки.

Создание такого генератора (рис. 3.2.) помогает уже на первом этапе исследования заложить основные принципы, проанализировать возможности и выработать качества ИРИИС, которые будут необходимы при обработке данных РВОИС в реальных условиях.

Программа данного генератора содержит следующие блоки:

1. Общие установки
2. Конструктор траекторий движения
3. Конструктор объектов
4. Конструктор сценариев
5. Выборка

В каждом блоке этой программы реализуются настройки, которые позволяют моделировать предполагаемые ситуации с разными классами движущихся объектов, а также менять состояния самой РИС в зависимости от заданных внешних условий, что обеспечивает репрезентативность выборки примеров для более качественного обучения нейронных сетей.

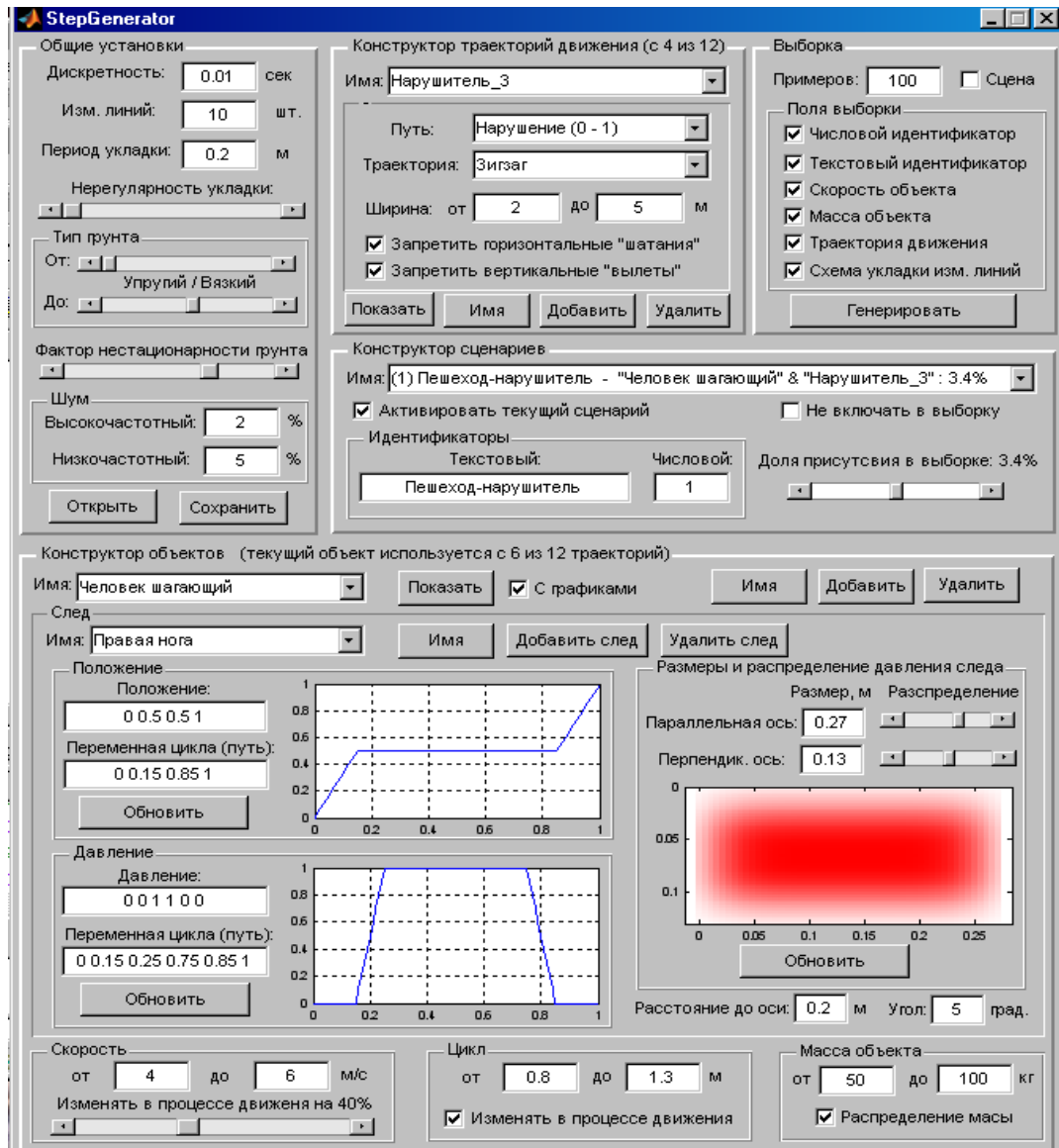


Рис. 3.2. Экранный интерфейс генератора сенсорных данных РИС

Рассмотрим функциональные возможности и настройки блоков генератора подробней.

### **Блок 1 «Общие установки»**

В панели инструментов блока «Общие установки» представлены следующие настройки:

- **Дискретность** – установление временного периода в секундах, через который осуществляются фиксация данных с измерительных линий.
- **Изм. линий** – установление количества уложенных оптоволоконных измерительных линий.



- **Период укладки** – изменение шага укладки измерительных линий (расстояние между двумя соседними измерительными линиями).
- **Нерегулярность укладки** – установление наклона измерительных линий на некоторый произвольный угол относительно друг друга. Таким образом, при моделировании есть возможность учитывать нерегулярность укладки измерительных линий в РИС (для упрощения расчетов сохраняется линейность измерительных линий).
- **Тип грунта** – регулирование упругости/вязкости грунта (подложки), на который уложено оптоволокно. В качестве примера упругого грунта (подложки) можно привести «резину», а вязкого – «глину». В первом случае, после снятия воздействия, оптоволокно под действием подложки (грунта) вернется к исходному состоянию (а соответственно к исходному состоянию вернутся и показания с измерительной линии), а, во втором случае, вязкая «глина» не даст оптоволокну (и соответственно, показаниям с него) отыграть обратно. С учётом того, что на протяжении зоны мониторинга могут одновременно существовать зоны с большей и меньшей упругостью/вязкостью, этот параметр задаётся как диапазон, который определяется положением пары бегунков: верхнего "От" и нижнего "До". Крайнее левое положение бегунков соответствует случаю максимальной упругости, а крайнее правое – максимальной вязкости. Положение этих бегунков и расстояние между ними устанавливает ширину и положение диапазона допустимого изменения этого параметра. Каждая точка пространства может иметь своё значение параметра упругости/вязкости, но не выходящее за рамки выбранного диапазона.
- **Фактор нестационарности грунта** – установление параметра, который приближает результаты математического моделирования к реальным данным. С его помощью в результаты численного моделирования вводится влияние целого ряда факторов, которые явно не учитываются при моделировании, но присутствуют в реальности, и, как правило, имеют

случайную природу (камни, корни деревьев и т.д.). Влияние этих побочных факторов на практике естественно осложняет идентификацию объекта. По этой причине значение параметра "Фактор нестационарности грунта", по возможности, выбирается большое при моделировании, что способствует разработке более реалистичной и надежной системы идентификации.

- **Шум** – установление зашумлённости сигналов, которое, как правило, неизбежно присутствует в измерительных каналах реальной ИС. В данном случае шум разделен по частотной составляющей. Высокочастотный шум – это погрешность текущего измерения, которая случайно изменяется от одного измерения к другому, а поэтому проявляется на частоте дискретизации всей системы. Низкочастотный шум может быть вызван нагревом/остыванием измерительных устройств (например, под действием солнца/ветра), "плаванием" питающего напряжения и т.д. Амплитуду этих шумов в процентах от амплитуды "номинального" сигнала можно задать порознь. Значения, которые приняты по умолчанию (высокочастотный - 2% и низкочастотный - 5%), вполне достаточны, но для повышения надёжности системы идентификации можно выбирать их выше установленных.

Загрузить и сохранить настроечные параметры моделирования этого и всех остальных блоков генератора позволяют кнопки "Открыть" и "Сохранить". Необходимо отметить, что все начальные установки, которые появляются на экране при запуске программы, также загружаются с диска. Для этого на диске присутствует обязательный файл default.mat. Если внести какие-то изменения в настройки моделирования, и нажав на кнопку "Сохранить" записать под именем default.mat, то при запуске программы в следующий раз по умолчанию уже будут присвоены новые настройки.

### **Блок 2 «Конструктор траекторий движения»**

В панели инструментов блока «Конструктора траекторий движения» представлены следующие настройки:

- **Имя** – название текущей траектории, т.е. той, параметры которой представлены в текущий момент на экране. Необходимо отметить, что при изменении этих параметров, их новые значения запоминаются и ставятся в соответствии с этим именем траектории автоматически, т.е. отдельной кнопки "Принять" не предусмотрено.
- **Путь** – задание "точки входа и выхода" объекта. Здесь предусмотрены четыре варианта:
  - <Нарушение 0-1> - пересечение границы по направлению к "нашей" территории;
  - <Побег 1-0> - пересечение границы по направлению к "чужой" траектории;
  - <Свой гуляет 0-0> - выход в зону мониторинга с нашей стороны границы, разворот в зоне мониторинга и возврат обратно на нашу сторону;
  - <Чужой гуляет 1-1> - выход в зону мониторинга с чужой стороны границы, разворот в зоне мониторинга и возврат обратно на чужую сторону.

В цифровых комментариях к этим позициям используются цифры 0 и 1, обозначающие точки входа/выхода на своей и чужой стороне границы соответственно.

- **Траектория** – задание "сложности" траектории движения, описываемые функциями, имеющими конечные первые и вторые производные во всех точках. Фактически при реализации этих траекторий в программе используются степенные полиномы, а четыре предлагаемых "уровня сложности" соответствуют порядкам используемых полиномов (от 1 до 4).

Например, для четвертого порядка эти полиномы будут иметь вид:

$$\begin{aligned} x(t) &= a_0 + a_1t + a_2t^2 + a_3t^3 + a_4t^4, \\ y(t) &= b_0 + b_1t + b_2t^2 + b_3t^3 + b_4t^4, \end{aligned} \quad (3.1.1)$$

где  $t$  изменяется от 0 до 1.

- **Ширина** – параметр, который определяет ширину пересечения границы в горизонтальном направлении (по координате  $x$ ), если пересечение границы происходит в вертикальном направлении (по координате  $y$ ). Для каждого отдельно примера значение этого параметра выбирается случайным образом из диапазона, заданного пунктами "от" и "до" (задается в метрах).

Наличие пунктов "Запретить горизонтальные шатания" и "Запретить вертикальные вылеты" обусловлено выбранным способом генерации траекторий, который не исключает возможности выхода за вертикальные (определяемые числом и шагом укладки измерительных линий) и горизонтальные (задаваемые параметром "Ширина") границы. Поставив галочки в соответствующих пунктах эти "вылеты" можно запретить.

Кнопка "Показать" необходима для просмотра примера сконструированной траектории. Для добавления новых и удаления существующих траекторий используют кнопки "Добавить" и "Удалить" соответственно. Созданная кнопкой "Добавить" траектория наследует все настроечные параметры той траектории, которая в момент создания была текущей. Для того чтобы изменить имя текущей траектории предусмотрена кнопка "Имя".

### **Блок 3 «Конструктор объектов»**

Пункт "Имя", а также верхняя строка кнопок "Показать", "Имя", "Добавить" и "Удалить" выполняют здесь те же функции что и соответствующие пункт и кнопки раздела "Конструктор траекторий", только здесь они относятся к конструированию объектов. Галочка "С графиками" позволяет на время конструирования объекта отключить визуализацию графиков данных с измерительных линий, что ускоряет процесс отладочного моделирования.

Принцип моделирования объектов таков, что движение объекта по заданной траектории состоит из повторяющихся циклов. Для этого длина всей траектории движения объекта делится на части, равные длине цикла (длина цикла задается в метрах, обозначим ее буквой  $h$ ). Внутри каждого из образовавшихся участков

(циклов) переменная цикла изменяется пропорционально пройденному на этом участке пути от 0 (в самом начале) до 1 (в самом конце), после чего на следующем цикле все повторяется. Обозначим переменную цикла буквой  $z$  ( $z \in [0,1]$ ). Пройденное объектом расстояние, от которого текущий цикл ведет свой отчет, обозначим  $S_0$ . Таким образом, расстояние, пройденное объектом к текущему моменту, всегда равно  $S_0 + z * h$ .

Конструирование объекта состоит в конструировании оставляемых им следов. Положение центра следа на длине траектории движения объекта определяется формулой (вид в программе):

$$S = S_0 + p(z) * h, \quad (3.1.2)$$

где  $p(z)$  – функция, определяющая положения следа в зависимости от переменной цикла. Интерфейс программы позволяет задавать эту функцию таблично как кусочно-линейную функцию. Фактическое положение следа сдвигается от траектории перпендикулярно её касательной на величину, задаваемую в пункте "Расстояние до оси". Это, в частности, позволяет задавать левую и правую ногу шагающего человека.

Аналогично вычисляется сила, с которой данный след (точка опоры) давит на грунт (вид в программе):

$$N = n(z) * m, \quad (3.1.3)$$

где  $n(z)$  – функция, определяющая силу давления в зависимости от переменной цикла,  $m$  – величина, пропорциональная массе объекта. Функция  $n(z)$  также задается таблично, в виде кусочно-линейной функции (настроечные параметры вкладки "След").

Назначение пункта "Имя", а также роль кнопок "Имя", "Добавить след" и "Удалить след", подобны таким же что были описаны выше.

В панели инструментов блока «Конструктор объектов» также представлены следующие настройки:

- **Положение** – выбор вида функции положения  $p(z)$  ("Переменная цикла (путь)" задаёт переменную по оси  $x$ , а "Положение" определяет соответствующие им значения функции по оси  $y$ ).
- **Давление** – выбор вида функции давления  $n(z)$  ("Переменная цикла (путь)" задаёт переменную по оси  $x$ , а "Давление" определяет соответствующие им значения функции по оси  $y$ ).
- **Размеры и распределение давления следа** – задание размеров и распределения давления.

Здесь стоит только отметить, что под параллельной осью понимается направление, совпадающее с направлением движения объекта, а под перпендикулярной осью, соответственно направление, перпендикулярное направлению движения объекта.

- **Расстояние до оси** – задание расстояния между центром следа и траекторией движения объекта.
- **Угол** – задание угла поворота следа вокруг своей оси (например, для моделирования эффекта "косолапости" у человека).
- **Распределение массы** (вкладка "Масса объекта") – распределение массы объекта между опорами при моделировании.

В формуле для вычисления давления (3.1.2) не учитывают распределение массы объекта между опорами. Выставление галочки в пункте "Распределение массы" вводит дополнительно этот эффект (механизм активизируется только тогда, когда суммарная масса по опорам превышает номинальную, т.е. ограничивает массу только сверху). Однако могут возникнуть случаи, когда механизм перераспределения массы не позволит смоделировать "эффект увеличение массы", например, воздействие на оптоволокно в результате прыжка. В этом случае галочку в пункте "Распределение массы" следует снять, и распределять массы между опорами в "ручном режиме", за счет соответствующего выбора функции  $n(z)$ .

Такие характеристики объекта как "Скорость", "Цикл", "Масса объекта" задаются в диапазонах. Выбор конкретных значений осуществляется случайным образом. Скорость может плавно изменяться в процессе движения объекта (в сторону ее уменьшения). Диапазон ее допустимого изменения задается в процентах бегунком "Изменять в процессе движения" (при 100% скорость может упасть до 0). Длина цикла может быть как постоянной в процессе всего движения, так и изменяться от цикла к циклу в установленном диапазоне, что задается с помощью галочки "Изменять в процессе движения".

#### **Блок 4 «Конструктор сценариев»**

В панели инструментов блока «Конструктор сценариев» представлены следующие настройки:

- **Активировать текущий сценарий** – активизация текущего сценария.  
Под текущим сценарием понимается текущий объект и его текущая траектория (т.е., что в данный момент находятся на экране). Устанавливая галочку в "Активировать текущий сценарий" в программе моделируется текущий сценарий "объект-траектория" при генерации выборки. Если галочка в этом пункте стоит, то становятся активными остальные настройки этого блока.
- **Идентификаторы** (числовой и текстовый) – информация, призванная идентифицировать тип нарушителя границы.  
Эта информация используется для эталонных выходов разрабатываемой экспертной системы. Группировка объектов в надежно разделяемые классы одна из основных задач, которую придется решать на этапе формирования выборки. Здесь текстовый идентификатор служит в первую очередь для удобства человеческого восприятия, а цифровой для удобства машинной обработки. Отсутствие одного или даже обоих идентификаторов не является критической ошибкой, препятствующей возможности генерации выборки.

- **Доля присутствия в выборке** – регулирование доли присутствия разных примеров в сгенерированной выборке.

Разрабатываемая система должна адекватно реагировать на всякие нештатные ситуации. Однако большое присутствие таких примеров в выборке данных, при ее статистической обработке может привести к тому, что полученное решение, стремящееся в равной степени удовлетворить всем примерам выборки, будет избыточно хорошо ориентироваться в нештатных ситуациях за счет того, что снизится точность идентификации штатных примеров. В статистике эта проблема хорошо известна. Выборки, с количественным перекосом одних примеров над другими называют нерепрезентативными. Параметр "Доля присутствия в выборке" позволяет снизить долю присутствия в выборке таких нештатных примеров. По умолчанию выполняется равное присутствие примеров каждого сценария в выборке.

Пункт "Не включать в выборку" позволяет временно исключить сценарий из выборки без потери его настроек (в случае исключения из выборки с помощью снятия галочки в пункте "Активировать текущий сценарий" все настройки потеряются).

Имя сценария формируется автоматически, по следующей схеме: ("Числовой идентификатор") "Текстовый идентификатор" – "Имя объекта" & "Имя траектории": "Доля присутствия в выборке"

### **Блок 5 «Выборка»**

В данном блоке реализованы настройки:

- **Примеров** – число примеров в выборке.
- **Сцена** – выбор режима генерации выборки.

Если в этом пункте поставить галочку, то при генерации выборки будет использоваться только текущий сценарий (даже в том случае, если он не активен в "Конструкторе сценариев").



- **Поля выборки** – именно те данные, которые наряду с данными с измерительных линий будут присутствовать в выборке. Основное назначение этих данных – эталонный выход экспертной системы.

Результаты генерации выборки записываются в \*.mat файл. Этот файл содержит не только сгенерированную выборку (структура с именем V), но и установки, при которых она была сгенерирована. Это позволяет загружать файл с выборкой так же, как и файл настроечных параметров, через кнопку "Открыть" в разделе "Общие установки". Эта возможность позволяет в случае необходимости просмотреть установочные данные, при которых была сгенерирована выборка.

### **3.1.2 Способы сжатия данных для формирования моделей образов по сгенерированным данным**

Одной из основных проблем, возникающих при функционировании ИРИИС на основе РВОИС, является возрастающий поток данных, который необходимо обрабатывать. Подбор преобразования для снижения объёма (сжатия) обрабатываемых массивов данных без существенной потери в них информативности, позволяет ускорить процесс распознавания динамических образов [155].

Для создания алгоритмов сжатия данных по сгенерированным данным использовали два способа. Первый способ основан на выделении моментов времени амплитудных всплесков и непосредственно самих амплитуд этих всплесков воздействий на ИЛ, а второй на формировании набора характерных признаков регистрируемого объекта по его воздействиям на РИС в процессе опроса системы. В первом случае, поток данных, поступающий с каждой измерительной линии, может быть компактно представлен в виде фиксированного объема информации о том, как давно состоялись последние  $m$  воздействий на измерительную линию и какой амплитуды они были. Этот способ подразумевает, что каждый новый регистрируемый всплеск приводит к "забыванию" информации о самом старом его предшественнике в ряду из  $m$  элементов. Форма сигнала при всплеске и его спадающий фронт имеют во многом вероятностную природу, а

поэтому являются не столь информативными. Кроме того, необходимо также учитывать, что объект при движении может в разные моменты времени оказать несколько воздействий на одну измерительную линию.

На рисунке 3.3. представлена графическая иллюстрация к предлагаемому способу (здесь и далее  $m = 4$ ;  $T_0$  – время инициализации). Аналогичные преобразования выполняются по всем  $n$  измерительным линиям РИС. В итоге образуются, и постоянно эволюционируют  $n$  векторов  $U_1, \dots, U_n$ , которые объединяясь, передаются в качестве входного вектора нейронной сети.

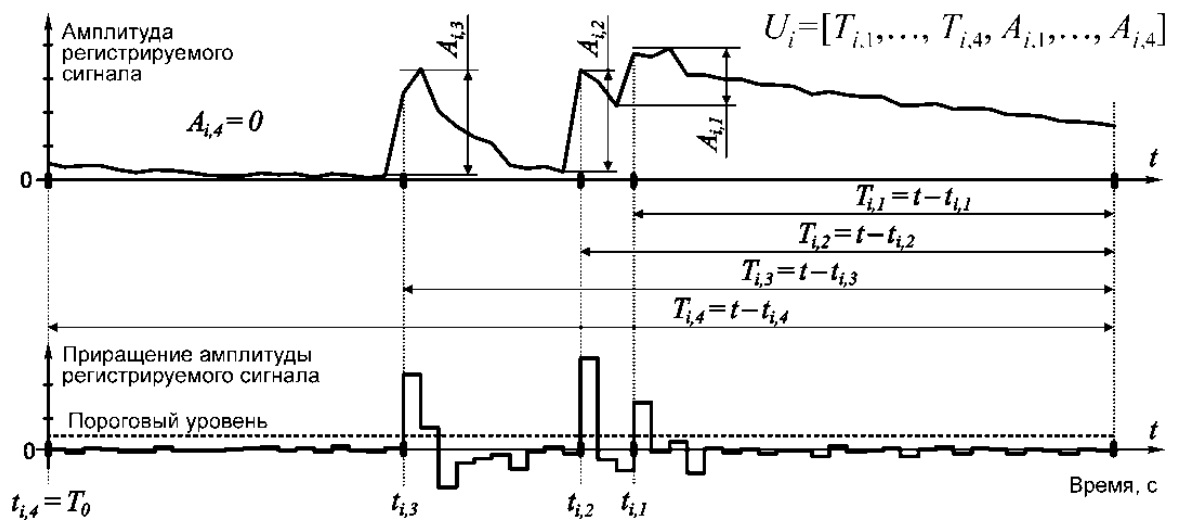


Рис. 3.3. Иллюстрация к способу компактного представления данных

Во втором случае, по полученным данным выделяют характерные признаки, по которым после обработки ИРИИС можно восстановить параметры самого регистрируемого объекта. Такой способ требует понимания как основных особенностей поведения (состояний) идентифицируемого объекта, так и знание того, какие параметры должны восстанавливаться в процессе распознавания ИРИИС.

Для реализации способа, основанного на выделении характерных признаков, необходимо иметь, кроме непосредственно самих сгенерированных данных измерительной сети  $V(n).Data$ , результат предобработки с целью выявления фактов воздействия на измерительные линии, то есть необходимо определить моменты времени, в которые на неё оказано воздействие. Эти моменты

определяются по факту положительного приращения амплитуды, превышающего исходно заданный порог. Важно знать в какие моменты объект «наступал» на измерительные линии. Измерительные линии, на которых регистрируется хотя бы одно внешнее воздействие, будем считать «активными». А измерительные линии, на которых за время инцидента не наблюдают внешних воздействий объекта, называют «пассивными». Время инцидента – это время регистрации одного примера или одного множества данных  $V(n)$ , в частности, для одного массива  $V(n).Data$ . Таким образом, формируется матрица признаков по всем примерам инцидентов, полученным с помощью генератора сенсорных данных РИС, и передаётся на вход нейронной сети.

Для численных экспериментов с применением обоих способов сжатия данных и последующего получения выборок примеров для различных видов движения идентифицируемых объектов использовался генератор сенсорных данных распределённой измерительной системы (РИС). Сенсорная зона РИС представлена в виде совокупности независимых измерительных линий, уложенных параллельно с некоторым шагом. Такого типа укладка ИЛ возможна при организации маскируемых волоконно-оптических охранных систем. Поэтому в основу моделирования результатов функционирования измерительной системы были положены принципы, наиболее существенные при реализации такого рода систем охраны периметра. К ним относятся: давление на грунт в районе прокладки оптоволоконной линии, фактор упругости грунта, особенности воздействия в зависимости от типа, скорости, траектории движущегося объекта. Конечный вид регистрируемого сигнала определяется с учетом существенного вклада случайного фактора, имитирующего широкий спектр неучтенных при моделировании и часто случайных явлений, неизбежно присутствующих в реальных условиях (степень однородности грунта, глубина прокладки оптоволоконной линии, температурные флуктуации, вид обуви нарушителя, его походка и т.д.). В последующих экспериментах рассматриваются 11 примеров условных нарушителей периметра, разбитых на 4 класса. Информация о них сведена в

таблицу 3.1. Масса, скорость и траектория нарушителя при моделировании каждого нового инцидента выбираются случайным образом.

Таблица 3.1. Сведения об условных нарушителях периметра

Условное обозначение класса	Представители	Траектория движения	Диапазон масс, кг	Диапазон скоростей, м/с
"Человек"	Человек: идущий, бегущий, хромающий	линейная, дугообразная	40 – 120	0,6 – 5
"Авто"	Автомобиль, трактор	линейная	800 – 2000	2 – 15
"ВелоМото"	Велосипед, мотоцикл, мотоцикл с коляской	линейная	50 – 250	2 – 10
"Фауна"	Собака, корова, заяц	линейная, дугообразная, зигзагообразная	5 – 250	0,5 – 4

Пример моделирования идущего человека и регистрируемых при этом сигналов представлен на рисунке 3.4.

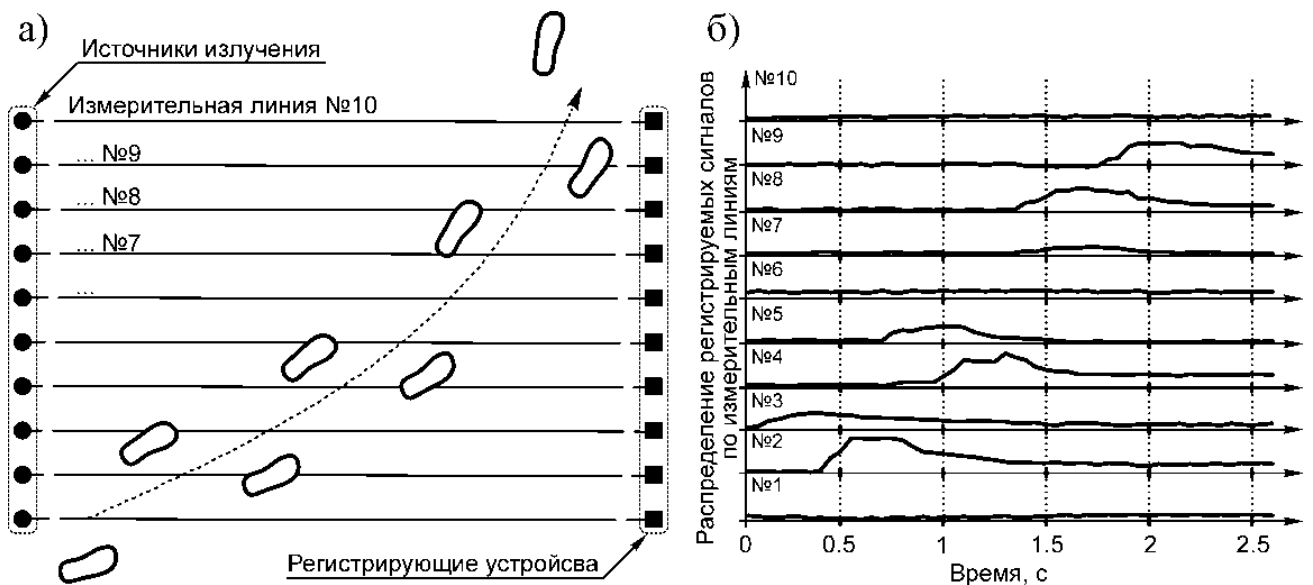


Рис. 3.4. Пример моделирования идущего в зоне мониторинга человека

Измерительная часть системы (рис. 3.4а) состоит из измерительных линий  $n = 10$ ; шаг укладки  $h = 20$  см. Частота регистрации данных:  $f = 20$  Гц. Рисунок 3.4б иллюстрирует пример получаемых данных, на основе которых система должна принять решение о классе нарушителя, его скорости, массе и направлении движения.

При формировании модели образа по признакам для реализации второго способа сжатия данных были выделены следующие **характерные признаки** (таблица 3.2.):

Таблица 3.2. Признаки для распознавания движущегося объекта по сгенерированным данным

Символы	Признаки
P1	Длительность инцидента
P2	Число активных измерительных линий
P3 P4	Среднее арифметическое и среднеквадратичное отклонение количества воздействий на активные измерительные линии
P5 P6	Среднее арифметическое и среднеквадратичное отклонение максимальных амплитуд для активных измерительных линий
P7 P8	Периодичность внешних воздействий на измерительную систему
P9-P12	Траектория движения нарушителя
P13	Среднеквадратичная ошибка аппроксимации траектории движения нарушителя

1. P1 – *длительность* инцидента, т.е. время между первым  $T_s$  и последним  $T_f$  воздействием на всю измерительную систему. Формула для расчёта:  $P1 = T_f - T_s$ ;
2. P2 – число *активных* измерительных линий ( $nLA$ );
3. P3 – среднеарифметическое значение и P4 - стандартное отклонение *числа воздействий* на активных измерительных линиях (*число воздействий* - сколько раз «наступили» на отдельно взятую измерительную линию);

4. P5 и P6 – среднеарифметическое значение и стандартное отклонение *максимальных амплитуд* на активных измерительных линиях, соответственно (Элементы вектора максимальных амплитуд вычисляются как  $A(i) = \max(V(n).Data(j, :))$ , где  $n$  – номер инцидента,  $i$  – порядковый ( $i = 1, 2, \dots, nLA$ ) и  $j$  – фактический номера активных ИЛ);
5. P6 и P7 – *периодичность* внешних воздействий, т.е. среднее приращение времени между соседними воздействиями и стандартное отклонение приращений времени на измерительную систему, соответственно.
6. Признаки *траектории* движения нарушителя формируются в виде коэффициентов степенного полинома, которые вычисляются в программе следующим образом:

```
T0 = T - min(T);    % Эта строчка гарантирует, что отсчет времени в T0 всегда будет
                   % вестись от нуля (min(T0) = 0). В противном случае, одна и та же
                   % траектория, но сдвинутая по оси времени, будет «выглядеть»
                   % по-разному.
k = polyfit(T0,Y,p), % Вычисляем коэффициенты степенного полинома.
```

где  $T$  – вектор моментов времени, в которые на измерительную систему оказано воздействие,  $Y$  - соответствующий этим моментам вектор номеров изм. линий (на которые оказано воздействие),  $p$  – порядок степенного полинома, величина которого должна быть меньше, чем число точек, по которым определяются коэффициенты.

Для вычисления  $p$  можно воспользоваться следующим алгоритмом:

```
if length(T) < 4
    p = length(T) - 1;
else
    p = 3;
end;
```

Результатом аппроксимации является функция следующего вида в программе:

$f(T0(i)) = a_0 + a_1 * T0(i) + a_2 * T0(i)^2 + a_3 * T0(i)^3 \sim Y(i)$ ,  
где  $a_0, \dots, a_3$  – искомые коэффициенты  $k=[a_3 \ a_2 \ a_1 \ a_0]$ .

Посмотреть результат аппроксимации можно с помощью функции:

```
plot(T0, Y, 'ro', [min(T0):0.01:max(T0)], polyval(k, [min(T0):0.01:max(T0)]))
```

Дальше необходимо выполнить следующий код:

```
a = k(end:-1:1); % Переворачиваем порядок следования элементов вектора k, т.к.
                % по умолчанию порядок от старших к младшим k=[a3 a2 a1 a0],
                % что неудобно.

a(5) = 0; % Эта строчка гарантирует, что вектор k будет физически содержать
          % необходимое число элементов (по умолчанию размерность вектора k
          % равна p+1, соответственно обращение к элементам, чей индекс больше
          % p+1 приведет к ошибке)

P9 = a(1); % коэффициент a0;
P10 = a(2); % коэффициент a1;
P11 = a(3); % коэффициент a2;
P12 = a(4); % коэффициент a3.
```

7. P13 - среднеквадратичная ошибка аппроксимации *траектории* движения нарушителя вычисляется в программе выражением:

$$P13 = \text{mse}(Y - \text{polyval}(k, T0)),$$

где обозначения переменных соответствуют предыдущему пункту.

Признаки выделяют с помощью программы, выполненной в среде MATLAB (листинг представлен в приложении №1).

### 3.1.3 Модуль преобразования реальных данных с макета ВОИС охраны периметра

На втором этапе исследований, с целью проверки возможностей и анализа разработанных способов использовались реальные данные, полученные с помощью макета ВОИС для маскируемых систем охраны периметра.

С помощью программного модуля преобразования данных, используя 200 примеров реально измеренных данных для различных вариантов поведения движущихся объектов двух типов (катящиеся и шагающие), было дополнительно получено 10000 альтернативных примеров, изменяя основные характеристики движущегося объекта (скорость, ускорение, массу и направление движения объекта), что обеспечило репрезентативность выборки примеров при обучении нейронной сети.

Этот программный модуль включает в себя следующие основные разделы интерфейса:

- **"Панель управления"** позволяет выбрать для рассмотрения 1 из 200 имеющихся примеров, указав класс объекта, направление его движения и номер примера от 1 до 50 в образовавшейся подгруппе.
- **"Фильтрация / Децимация"** устанавливает параметры фильтрации (используется скользящее среднее заданного порядка; 1 – не фильтровать) и один из вариантов "прореживания" данных: **"Децимация"** - брать каждый  $n$ -й отсчет исходных временных рядов (итоговая частота дискретизации:  $2500/n$  Гц); **"Фиксированная размерность"** – приводить размерность данных к указанной величине (используется линейная интерполяция).

Коэффициент децимации  $n$  может быть дробным числом. При значениях коэффициента децимации  $n < 1$  режим децимации автоматически отключается.

- **"Область данных"** – раздел, где указывается рассматриваемая область данных. Допускаются три варианта:

**"Весь диапазон"** – использовать весь доступный временной отрезок исходных данных;

**"Заданный"** – пользовательская область, диапазон которой можно индивидуально выбрать для каждого примера, нажав на кнопку **"Интервал"** (два раза щелкнуть мышкой в поле основного графика, расположенного ниже, в любом порядке указав левую и правую границу);

**"Интервал активности"** – рассматривать только интервал времени, на котором объект проявляет активность.

Пользовательская область должна содержать весь интервал активности. При попытке нарушить это требование программа автоматически приведет пользовательскую область к соответствующим границам интервала активности.

При изменении диапазона пользовательской области он сразу же автоматически сохраняется в файле `ShowData.ini`.



Если кнопка *"интервал"* нажата случайно, то отказаться от выделения можно, если щелкнуть мимо области графика по координате  $y$ .

- **"Показывать"** – раздел, где можно указать отображение графиков изменения положения объекта относительно измерительных линий.

Отображению могут подлежать:

*"Теоретическая траектория"* – траектория, полученная по известной пространственной траектории объекта из предположения постоянства скорости его движения на интервале активности;

*"Расчетная траектория"* – траектория, полученная в результате обработки реальных экспериментальных данных.

- **"Воздействия"** включает или выключает подсветку областей активности по каждой измерительной линии. Здесь же расположены элементы управления, с помощью которых можно задавать (кнопка *"Указать"*) и удалять (кнопка *"Удалить"*) эти области. Для указания новой области активности необходимо на графике в окрестностях измерительной линии щелкнуть мышкой, последовательно и в любом порядке задав ее левую и правую границы. Если новая область перекрывает уже существующую, то эта существующая область заменяется новой. При удалении области необходимо один раз щелкнуть мышкой в ее окрестностях. Окрестностью измерительной линии номер  $i$  считается диапазон от  $i-0.5$  до  $i+0.5$  на графике ( $i = 1, \dots, 8$ ).

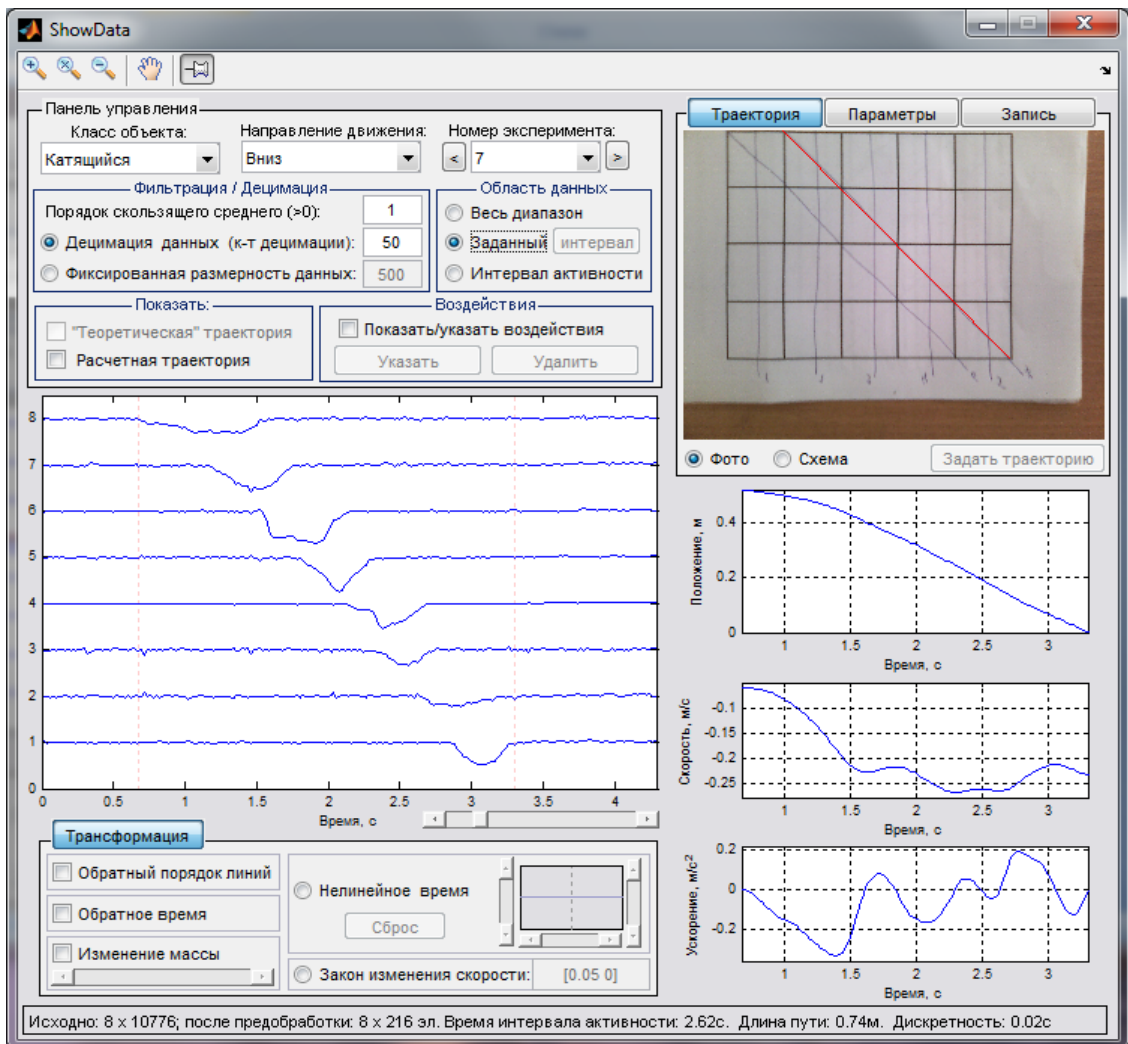
Вид модуля преобразования измеренных данных представлен на рисунке 3.5 (а, б) (рисунок 3.5(а) для катящегося объекта и рисунок 3.5(б) для шагающего). Все изменения областей активности сразу же автоматически сохраняются в файле `ShowData.ini`.

Главное *графическое окно* программы по умолчанию отображает данные по всем измерительным линиям, а так же указываются границы следующих диапазонов: пользовательский (задается кнопкой *"интервал"* в разделе *"Область данных"*), выделен цветом и ограничен светло-серыми пунктирными линиями;

интервал активности, формируемый автоматически по моментам первого и последнего воздействия на измерительные линии (области воздействий указываются / удаляются с помощью кнопок "Указать" / "Удалить" раздела "Воздействия"). Границы интервала активности отмечены светло-красными пунктирными линиями. Области по краям, не входящие в упомянутые границы, подсвечиваются серым цветом.

По желанию пользователя в графическое поле так же может быть выведена информация об областях активности по измерительным линиям и информация о траекториях движения объекта. Способ отображения траекторий движения объекта зависит от его класса.

a)



б)

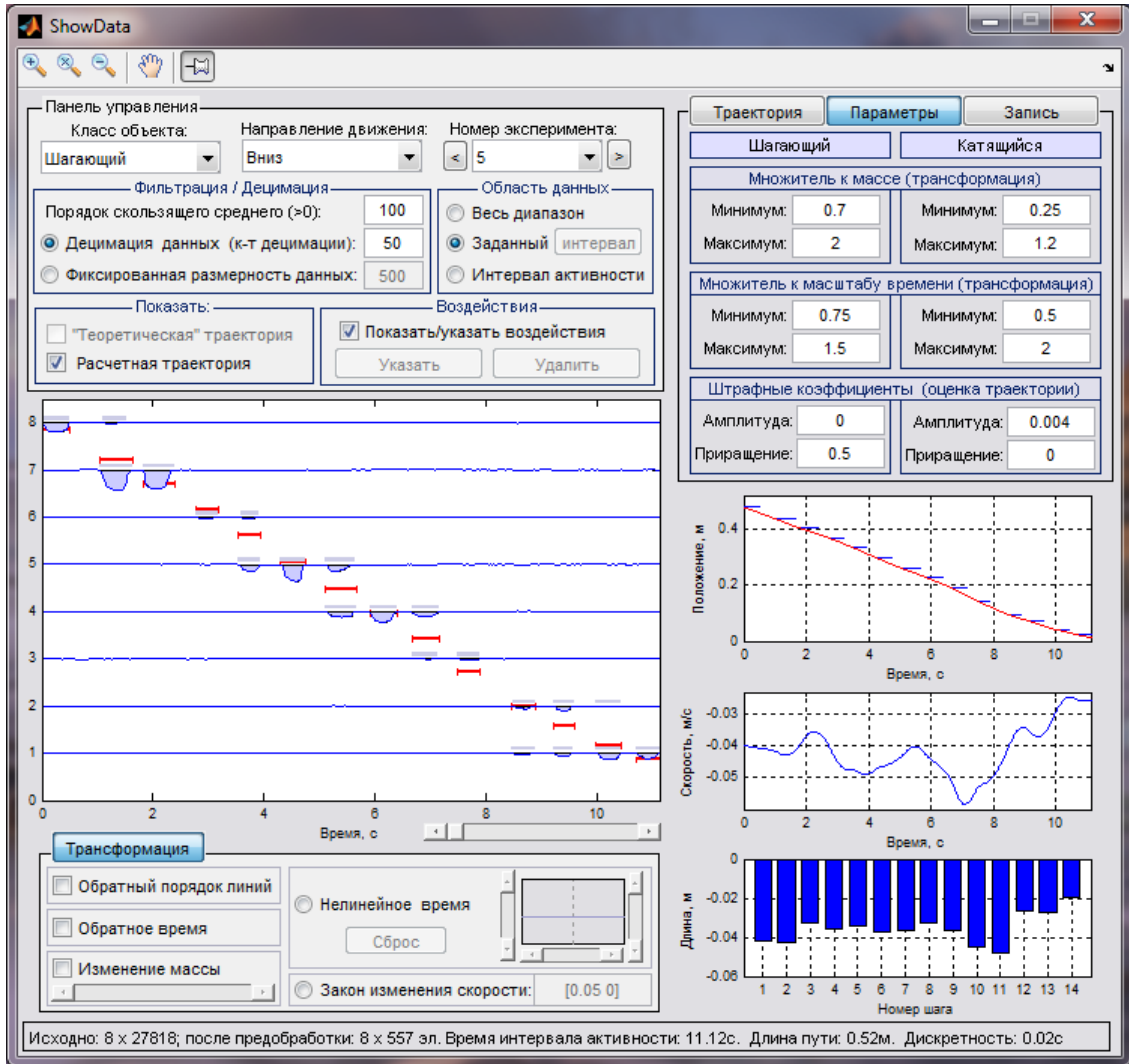


Рис. 3.5. Экранный интерфейс модуля преобразования измеренных данных: а) «катящийся объект»; б) «шагающий объект»

Для шагающего объекта визуализируются позиции и продолжительности шагов, для катящегося – отображается непрерывная траектория.

Три графика в правой части интерфейса программы представляют графики изменения положения объекта, его скорости, а так же, в зависимости от класса объекта, длины каждого из шагов для шагающих объектов, и ускорение - для катящихся представителей. Эта графическая информация формируется путем анализа фактических данных примера с учетом информации об областях воздействий, которая задается пользователем (кнопки "Указать" / "Удалить" в разделе "Воздействия") и сохраняется в файле ShowData.ini. График

положения и график, накладываемый на основную область данных по команде *"Показать/Расчетная траектория"* – одинаковы, и отличаются только масштабом.

Отмечая такие категории как положение и скорость, везде речь идет об их проекциях лишь на одну координату, направленную перпендикулярно ориентации измерительных линий (положение, скорость и ускорение в поперечном направлении).

График непрерывного положения для шагающих объектов строится как сплайн-интерполяция дискретных положений объекта. График скорости образуется путем дифференцирования по времени кривой на графике положения. Скорость и положение на заключительном отрезке времени, когда сделан последний шаг, но объект еще находится в зоне мониторинга, формируются как результат линейной экстраполяции скорости движения на предпоследнем шаге.

Длина первого шага на графике длин шагов вычисляется как разница между краем чувствительной зоны (ширина 0,52м) и координатой точки первого касания.

Режим *"Трансформация"* служит для преобразования экспериментальных данных с целью имитации альтернативных примеров. Активация/деактивация соответствующих элементов управления выполняется включением/выключением кнопки *"Трансформация"*. В режиме "трансформации" становятся недоступными процедуры изменения границ пользовательской области и областей активности (соответственно, кнопки *"интервал"*, *"Указать"* и *"Удалить"* в разделе *"Панель управления"*), а также отключается возможность наложения *"Теоретической траектории"*.

Доступны следующие элементы трансформации:

- *"Обратный порядок линий"* – реализация обратного порядка измерительных линий, что приводит к имитации движения в обратном направлении;
- *"Обратное время"* – время "прокручивается" в обратном направлении, что приводит к имитации движения "задом наперед";

- *"Изменение массы"* – имитация изменения массы путем умножения "провала" амплитуды сигнала в областях воздействия на некоторый коэффициент, определяемый положением слайдера; а так же два способа нелинейного преобразования интенсивности течения времени.
- *"Нелинейное время"* – нелинейное искажение времени на интервале активности, приводящее к изменению интенсивности протекания процессов и визуальному растяжению по времени одних участков и сжатию других. Степень и закономерность сжатия/растяжения (ускорения/замедления времени) выбирается с помощью слайдеров и отображается на специальном графике. По оси абсцисс на этом графике отложено исходное время, а по оси ординат – степень масштабирования (вверх – ускорение, вниз – замедление). Горизонтальный уровень на графике указывает уровень единичного масштаба времени (отсутствие искажений). Вверх откладывается во сколько раз процесс необходимо ускорить, а вниз – во сколько раз замедлить. Фактически же закон масштабирования определяется многочленом второго порядка. Кнопка *"Сброс"* выводит слайдеры в состояние с постоянным единичным масштабом (отсутствие искажений).
- *"Закон изменения скорости"* – позволяет явно указать закон изменения скорости объекта, к которому с помощью нелинейной трансформации времени приводятся фактические данные примера. Закон задается в форме вектора, содержащего коэффициенты степенного полинома. Первый элемент его отвечает за старшую степень, а последний – за постоянный член. Пример: [0.5] – движение с постоянной скоростью 0,5 м/с, [0.2 0] – движение с нулевой начальной скоростью и ускорением 0,2 м/с<sup>2</sup>. Параметры задаются без учета направления движения объекта (заимствуется у исходных данных). Для корректной работы процедуры требуется монотонность траектории исходного движения объекта. (Процедура не

содержит проверок на корректность и реализуемость вводимых данных, а поэтому может приводить к ошибкам).

Вкладка **"Траектория"** отражает либо фотографию с выделенной на ней траекторией (**"Фото"**), либо ее схематичное представление (**"Схема"**). Кнопка **"Задать траекторию"** служит для ручного указания траектории движения объекта по точкам. В режиме формирования траектории: левая кнопка мыши – выбрать новую точку, правая – закончить процесс.

Изменение пространственной траектории движения объекта сразу же автоматически сохраняется в файле `ShowData.ini`.

Вкладка **"Параметры"** содержит две группы параметров, задаваемых отдельно для шагающих и катящихся объектов. Поля **"Множитель к массе"** и **"Множитель к масштабу времени"** определяют диапазоны изменения величин, задаваемых слайдерами в поле **"Трансформация"**. Эти же диапазоны используются для автоматической генерации примеров в процессе формирования выборок для записи. В поле **"Штрафные коэффициенты"** задаются значения штрафов и определяют функционалы, минимизируемые в процессе поиска траекторий по данным эксперимента (ограничения на амплитуду и ее приращение в правой части дифференциального уравнения, описывающего процесс движения объекта).

Вкладка **"Запись"** позволяет указать подгруппы примеров, используемые для записи, а так же число дополнительных (трансформированных) примеров и способ их формирования. Данные записываются в файл `*.mat` в виде структуры с именем `V`. Число элементов этой структуры соответствует числу примеров в сгенерированной выборке. Поля структуры и их описание представлены в таблице 3.3.

На кнопке **"Запись выборки"** отображается число примеров, которое будет сформировано при заданных установках. Кнопка **"Запись текущего примера"** осуществляет сохранение на диске того примера, который в текущий момент

отображен на экране. Объем и область записываемых данных выполняется согласно установкам разделов "Фильтрация / Децимация" и "Область данных".

Таблица 3.3. Расшифровка записываемых данных в файл

Поле структуры V	Описание [размерность переменной]
Class	Класс объекта (0 – шагающий; 1 - катящийся)
Direction	Направление движения с учетом трансформации данных (0 - "вверх"; 1 - "вниз")
Number	Номер примера в подгруппе (от 1 до 50)
Mass	Масса объекта [кг]
Data	Массив данных по всем измерительным линиям размерностью $8 \times L$ , где $L$ – число элементов во временных рядах
SampleTime	Интервал дискретности [с]
Start	Начало области активности (задается в дискретных отсчетах [дискр.])
Finish	Конец области активности [дискр.]
Position	Положение в поперечном направлении [м] (сплайн, аргумент в дискр.)
Velocity	Скорость в поперечном направлении [м/с] (сплайн, аргумент в дискр.)
Action	Для катящихся объектов - ускорение [м/с <sup>2</sup> ] (сплайн, аргумент в дискр.). Для шагающих – матрица размером $j \times 3$ , где $j$ – номер шага, Action( $j$ , 1) – положение в поперечном направлении [м] Action( $j$ , 2) – момент начала $j$ -го шага [дискр.] Action( $j$ , 3) – момент завершения $j$ -го шага [дискр.]
Act( $i$ ).s Act( $i$ ).f	Векторы с разметкой моментов начал (поле .s) и соответствующих им концов (поле .f) воздействий по каждой линии $i$ [дискр.]
Trajectory.x Trajectory.y	Векторы, описывающие пространственную траекторию движения объекта с учетом направления [м]
PathLength	Длина пути по траектории движения [м]
RightLim	Правый край пользовательской области [дискр.]
LeftLim	Левый край пользовательской области [дискр.]
InverseLine	При формировании примера инвертирован порядок измерительных линий [логическая переменная]
InverseTime	- инвертировано течение времени [логическая переменная]
VariableMass	- трансформирована масса объекта [логическая переменная]
NotLineTime	- выполнена нелинейная трансформация времени [логическая переменная]

Перед записью данные нормализуются, а поэтому их амплитуда отличается от амплитуды исходных данных в текстовых файлах. Нормализация выполняется по каждому примеру и каждой измерительной линии отдельно. Нормализация выполняется таким образом, чтобы среднее значение сигналов вне областей активности было равно 1.

**Строка инструментов** (пиктограммы вверху) содержит элементы управления графиками. "Лупа с крестиком" возвращает отображаемые области всех графиков к их исходным диапазонам. Кнопка "Кнопка" служит для защиты от случайного нажатия на элементы управления, отвечающие за указание пространственной траектории движения объекта (кнопка "*Задать траекторию*") и управление областями активности по каждой измерительной линии (кнопки "*Указать*" и "*Удалить*"). Для активации этих элементов управления необходимо отжать кнопку "Кнопка".

### 3.1.4 Способы сжатия данных для формирования модели образа по реальным данным с макета ВОИС охраны периметра

Предварительно полученные данные в виде временных рядов с ИЛ макета ВОИС охраны периметра проходили два этапа предобработки. На первом этапе происходит фильтрация скользящим средним с заданным периодом (или сглаживающим интервалом) и децимация исходных данных в модуле преобразования данных. На втором этапе по полученным двум скользящим средним с разными периодами  $n_1$  и  $n_2$ , а именно, по расхождению значений полученных функций в определённые моменты времени, устанавливаются непосредственно временные интервалы возникновения воздействий на каждой ИЛ по превышению разницы между возмущениями допустимого заданного шумового порога  $u_0$ . **Скользящее среднее** вычисляется по формуле:

$$SMA_t = \frac{1}{n} \sum_{i=0}^{n-1} p_{t-i}, \quad (3.1.4)$$

где  $SMA_t$  – значение простого скользящего среднего в точке  $t$ ,  $n$  – количество значений исходной функции для расчёта скользящего среднего (сглаживающий



интервал или период), чем шире сглаживающий интервал, тем более плавный график функции,  $p_{t-i}$  – значение исходной функции в точке  $t-i$ .

Из предыдущего своего значения *простое скользящее среднее* может быть получено по следующей рекуррентной формуле:

$$SMA_t = SMA_{t-1} - \frac{p_{t-n}}{n} + \frac{p_t}{n}, \quad (3.1.5)$$

где  $SMA_t$  – значение простого скользящего среднего в точке  $t$ ,  $SMA_{t-1}$  – предыдущее значение простого скользящего среднего в точке  $t-1$ ,  $p_{t-n}$  – значение исходной функции в точке  $t-n$  (в случае временного ряда, самое «раннее» значение исходной функции, используемое для вычисления предыдущей скользящей средней),  $p_t$  – значение исходной функции в точке  $t$  (в случае временного ряда, «текущее» – последнее значение).

После двух этапов предобработки данных одним из двух предложенных способов (методом главных компонент или выделением характерных признаков) формируют МО из главных компонент или признаков для последующего распознавания параметров идентифицируемого движущегося объекта. Опишем подробнее использованные способы сжатия данных.

*Метод главных компонент* применяют как метод редукции (сокращения) данных [155, 157]. Процедура выделения главных компонент подобна *вращению, максимизирующему дисперсию* исходного пространства переменных и сводится к вычислению собственных векторов и собственных значений ковариационной матрицы исходных данных или к сингулярному разложению матрицы данных. Ковариационная матрица случайного вектора – квадратная симметрическая матрица, на диагонали которой располагаются дисперсии компонент вектора, а недиагональные элементы – ковариации между компонентами. Для многомерной случайной величины  $\mathbf{X}$  сокращённая формула для вычисления матрицы ковариаций:

$$\text{cov}(\mathbf{X}) = \mathbb{E}[\mathbf{X}\mathbf{X}^T] - \mathbb{E}[\mathbf{X}] \cdot \mathbb{E}[\mathbf{X}^T]. \quad (3.1.6)$$

Векторы главных компонент для задач о лучшей аппроксимации и о поиске ортогональных проекций с наибольшим рассеянием – это ортонормированный

набор  $\{a_1, a_2, \dots, a_n\}$  собственных векторов эмпирической или выборочной матрицы  $C$ , расположенных в порядке убывания собственных значений  $\lambda$ :  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n \geq 0$ . Эмпирическая или выборочная ковариационная матрица имеет вид:

$$C = [c_{ij}], c_{ij} = \frac{1}{m-1} \sum_{l=1}^m (x_{li} - \bar{X}_i)(x_{lj} - \bar{X}_j). \quad (3.1.7)$$

Математическое содержание метода главных компонент – это *спектральное разложение* ковариационной матрицы  $C$ , то есть представление пространства данных в виде суммы ортогональных собственных подпространств  $C$ , а самой матрицы  $C$  – в виде линейной комбинации ортогональных проекторов на эти подпространства с коэффициентами  $\lambda_i$ . Если  $X = \{x_1, \dots, x_m\}$  – матрица, составленная из векторов-столбцов центрированных данных, то  $C = \frac{1}{m-1} X^T X$  и задача о *спектральном разложении* превращается в задачу о *сингулярном разложении* матрицы данных  $X$ . Число  $\sigma \geq 0$  называется *сингулярным числом* матрицы  $X$  тогда и только тогда, когда существуют **правый и левый сингулярные векторы**: такие  $m$ -мерный вектор-строка  $b_\sigma$  и  $n$ -мерный вектор-столбец  $a_\sigma$  (единичной длины), что выполнено два равенства:

$$X a_\sigma = \sigma b_\sigma^T; b_\sigma X = \sigma a_\sigma^T. \quad (3.1.8)$$

Пусть  $p = \text{rang} X \leq \min\{n, m\}$  – ранг матрицы данных. *Сингулярное разложение* матрицы данных  $X$  – это её представление в виде

$$X = \sum_{l=1}^p \sigma_l b_l^T a_l^T; X^T = \sum_{l=1}^p \sigma_l a_l b_l \quad (x_{ij} = \sum_{l=1}^p \sigma_l b_{li} a_{lj}), \quad (3.1.9)$$

где  $\sigma_l > 0$  – сингулярное число,  $a_l = (a_{li}), i=1, \dots, n$  – соответствующий правый сингулярный вектор-столбец, а  $b_l = (b_{li}), i=1, \dots, m$  – соответствующий левый сингулярный вектор-строка ( $l=1, \dots, p$ ). Правые сингулярные векторы-столбцы  $a_l$ , участвующие в этом разложении, являются векторами главных компонент и собственными векторами эмпирической ковариационной матрицы  $C = \frac{1}{m-1} X^T X$ , отвечающими положительным собственным числам  $\lambda_l = \frac{1}{m-1} \sigma_l^2 > 0$ . Хотя формально задачи сингулярного разложения матрицы данных и спектрального разложения ковариационной матрицы совпадают, алгоритмы вычисления

сингулярного разложения напрямую, без вычисления ковариационной матрицы и её спектра, более эффективны и устойчивы.

**Выделение набора характерных признаков** осуществляют для формирования из них массива признаков, который подаётся на вход нейронной сети, при определении основных параметров идентифицируемого движущегося объекта. Полный набор признаков представлен в таблице 3.4.

Таблица 3.4. Признаки для распознавания движущегося объекта по реальным данным

Символы	Признаки
P(1)	<i>длительность</i> инцидента
P(2)	число <i>активных</i> измерительных линий
P(3)	суммарное <i>количество воздействий</i> по всем линиям за время мониторинга ИС
P(4)	среднеквадратичное отклонение <i>числа воздействий</i> по линиям
P(5)	среднеарифметическое значение <i>амплитуд воздействий</i> по всем линиям
P(6)	среднеквадратичное значение <i>амплитуд воздействий</i> по всем линиям
P(7)	средняя <i>периодичность</i> воздействий
P(8)	среднеквадратичное отклонение <i>периодичности</i> воздействий
P(9)-P(12)	коэффициенты полинома, описывающего <i>траекторию</i> движения объекта
P(13)	среднеквадратичная ошибка аппроксимации <i>траектории</i> движения
P(14)	<i>непрерывность</i> воздействия на ИЛ
P(15)	<i>параллельность</i> воздействий на ИЛ
P(16)	<i>одновременность</i> воздействий на ВОИС
P(17)	<i>длина «пути»</i>
P(18)	<i>направление</i>
P(19), P(20)	взвешенные <i>«центры масс»</i> положения

Рассмотрим подробнее входящие в таблицу характерные признаки:

P(1) – *длительность* инцидента, т.е. время между первым и последним воздействием на всю измерительную систему;

P(2) – число *активных* измерительных линий, т.е. количество задействованных ИЛ за время инцидента ИС;

P(3) – накапливаемая сумма воздействий по всем измерительным линиям, т.е. суммарное количество воздействий по всем линиям за время инцидента ИС;

P(4) – среднеквадратичное отклонение числа воздействий по линиям;

P(5) – среднеарифметическое значение амплитуд воздействий по всем линиям;

P(6) – среднеквадратичное значение амплитуд воздействий по всем линиям (амплитуда воздействия – максимальная амплитуда воздействия, взятая относительно начала и конца времени воздействия);

P(7) – средняя периодичность воздействий (периодичность воздействий – время между последовательными воздействиями);

P(8) – среднеквадратичное отклонение периодичности воздействий;

P(9)-P(12) – коэффициенты полинома, описывающего траекторию движения объекта;

P(13) – среднеквадратичная ошибка аппроксимации траектории движения;

P(14) – непрерывность воздействия, т.е. относительная суммарная продолжительность самих воздействий на ИЛ к длительности инцидента в целом;

P(15) – параллельность воздействий, т.е. относительное суммарное количество воздействий на ИЛ к длительности инцидента в целом;

P(16) – одновременность воздействий, т.е. максимальное число линий с одновременным воздействием;

P(17) – длина «пути», т.е. сумма модулей приращения положения;

P(18) – направление, т.е. сумма знаков приращения направлений;

P(19), P(20) – взвешенные «центры масс» положения (взвешивание прямым и обратным временем).

Описанные выше способы реализованы в виде программ в системе MATLAB (листинг программ представлен в приложении №2). В М-файле Action\_Search.m находят временные интервалы возникновения воздействий

на каждой измерительной линии ВОИС. Процедуры для вычисления главных компонент и выделения характерных признаков введены как М-файлы – функции: `Make_Signs_PCA.m` и `Make_Signs.m`.

### **3.1.5 Результаты численного моделирования и обработки по сгенерированным данным с применением нейронных сетей**

Проверка работы созданного модуля обработки данных ИРИИС выполняется на выборках примеров, сформированных генератором сенсорных данных РИС, описанным в разделе 3.1.1. Проведённые численные эксперименты на сгенерированных данных с применением способов сжатия данных, рассмотренных в разделе 3.1.2, позволяют судить о качестве обработки предложенным комплексом программ и установить эффективность, корректность работы модуля при переменных условиях сбора данных и сложности параметрического задания модели распознаваемого динамического образа [79, 175, 179, 180]. Детализация отдельных характеристик идентифицируемого объекта, а также априорные данные, могут отразиться на качественных свойствах самой модели и повлиять на её чувствительность, адекватность и устойчивость.

На вход нейронной сети подаются объединённые и постоянно эволюционирующие  $n$  векторов  $U_1, \dots, U_n$ . (раздел 3.1.2) Для разделения объектов (или «условных нарушителей») на 4 класса (таблица 3.1.) используются 4 нейронные сети. Каждая из них обучалась возвращать 1 в случае принадлежности данных входного вектора к ее классу, и 0 в противном случае. Для определения направления движения используют одну нейронную сеть, обучаемую возвращать либо 1, либо -1, в зависимости от направления пересечения зоны мониторинга. В обоих случаях используются перцептронные структуры нейросетей 80-10-1-1, обучаемые по критерию минимума среднеквадратичной ошибки. В случае построения нейросетевых подсистем оценки скорости и массы «нарушителя», этот критерий обучения, стремящийся к равномерному распределению абсолютной ошибки по всем примерам обучающей выборки, приводит к критическим значениям относительной ошибки для «нарушителей» с малой

массой/скоростью. Для преодоления этой проблемы без изменения критерия качества обучения предварительно нормализовали эталонный выход в обучающей выборке, приведя его значения к одному порядку. В данном случае это реализовано посредством извлечения квадратного (для оценки скорости) и кубического (для оценки массы) корней. При работе с обученными таким образом нейронными сетями необходимо выполнять обратную операцию, т.е. возводить возвращаемые ими значения в квадрат и куб соответственно. Оценка скорости и массы объекта выполнена на базе перцептронных структур 80-10-1. Здесь и далее во всех случаях для обучения нейронных сетей используется метод Левенберга-Марквардта (LMA) [156, 157]. Трудности, возникшие на этапе обучения нейронных сетей, связаны с большой размерностью их входного вектора ( $2 \cdot m \cdot n = 80$  элементов), способствующей проявлению эффекта переобучения. Для преодоления этой проблемы возникла необходимость существенного повышения репрезентативности обучающей выборки. Для этого было выполнено моделирование 10000 инцидентов и на их базе сформирована обучающая выборка, содержащая 20000 примеров. Кроме того, 10% этих данных использовалось в качестве тестовой выборки для своевременной остановки процесса обучения.

На рисунках 3.6. и 3.7. представлены результаты тестирования нейронных сетей с помощью примеров, которые не участвовали в процессе обучения.

Рисунок 3.6. иллюстрирует типовые примеры обработки данных нейронными сетями при движении «нарушителя» в зоне мониторинга «в режиме реального времени».

На рисунке 3.7. представлена более полная картина зависимости точности экспертных оценок от нормированного времени движения «нарушителя» (0 соответствует моменту входа в зону мониторинга, 1 - выходу из нее). Из этого рисунка видно, что 40% времени движения нарушителя в зоне мониторинга достаточно для того, чтобы в 90% случаев его класс был определен верно. По результатам полного цикла перемещений в зоне мониторинга класс нарушителя

определяется правильно в 97% случаев, а его направление в 99% случаев. Наибольшая погрешность (>30%) регистрируется при оценке массы «нарушителя».

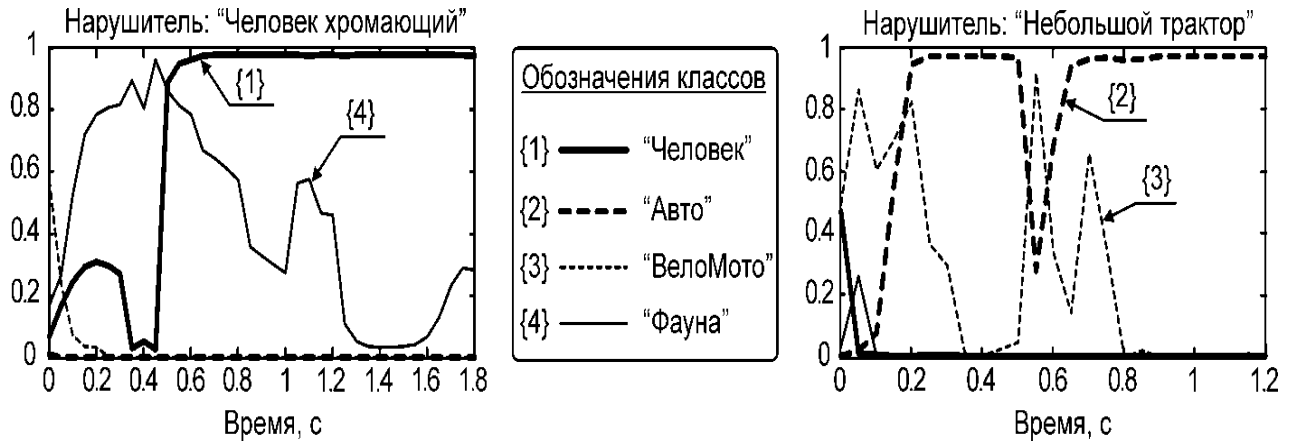


Рис. 3.6. Типовые примеры обработки данных нейронными сетями

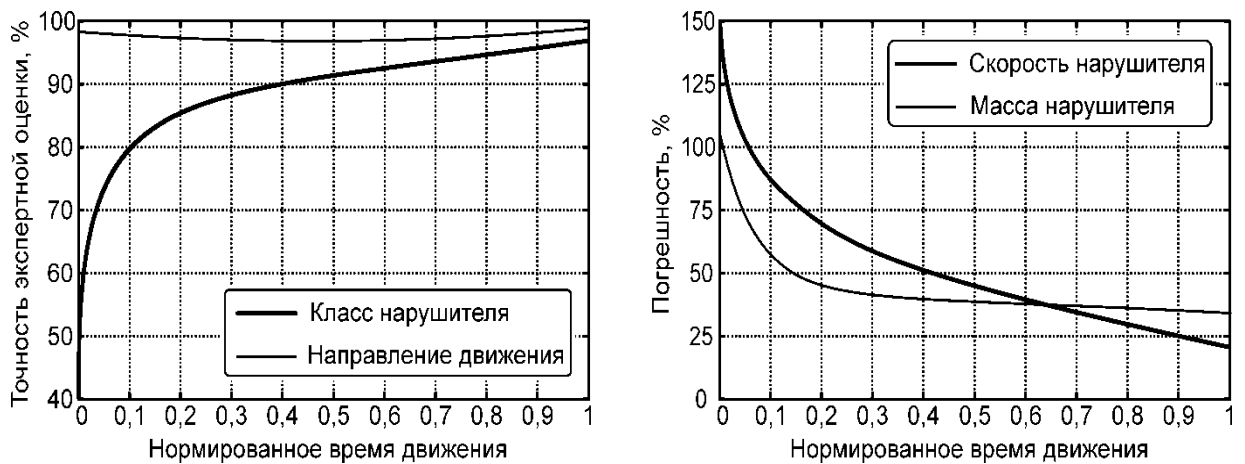


Рис. 3.7. Характеристики точности системы

Часть исследования была сосредоточена на проблеме локализации местоположения объекта. Суть предлагаемого приема для предложенного типа РВОИС (как маскируемой сети охраны периметра) состоит в разбиении периметра на участки, где каждому участку соответствует уникальная последовательность укладки измерительных линий (максимальное число участков с уникальной укладкой:  $z = n!$ , в данном случае  $z = 10! = 3\,628\,800$ ). «Нарушитель», преодолевая зону мониторинга, движется от одного ее края до другого. При этом он произвольно оказывает воздействия на измерительные линии в соответствии

с той последовательностью, с которой они физически уложены на данном участке. Таким образом, локализация участка нарушения может быть выполнена путем сопоставления известных последовательностей укладки измерительных линий с последовательностями, порожденными движением нарушителя.

Рассмотрим влияние нерегулярной последовательности укладки измерительных линий на способность системы классифицировать нарушителя, оценивать его массу и скорость. Исследуем наиболее жесткий случай – произвольная последовательность измерительных линий для каждого примера обучающей выборки. Во всем остальном проводимая серия экспериментов полностью повторяет ранее проведенную серию. Совмещенные результаты обеих серий экспериментов представлены на рисунке 3.8. Из этого рисунка 3.8 видно, что нерегулярность укладки в основном повлияла на способность системы к "раннему обнаружению". В конечном же счете, в обоих случаях достигаются соизмеримые результаты.

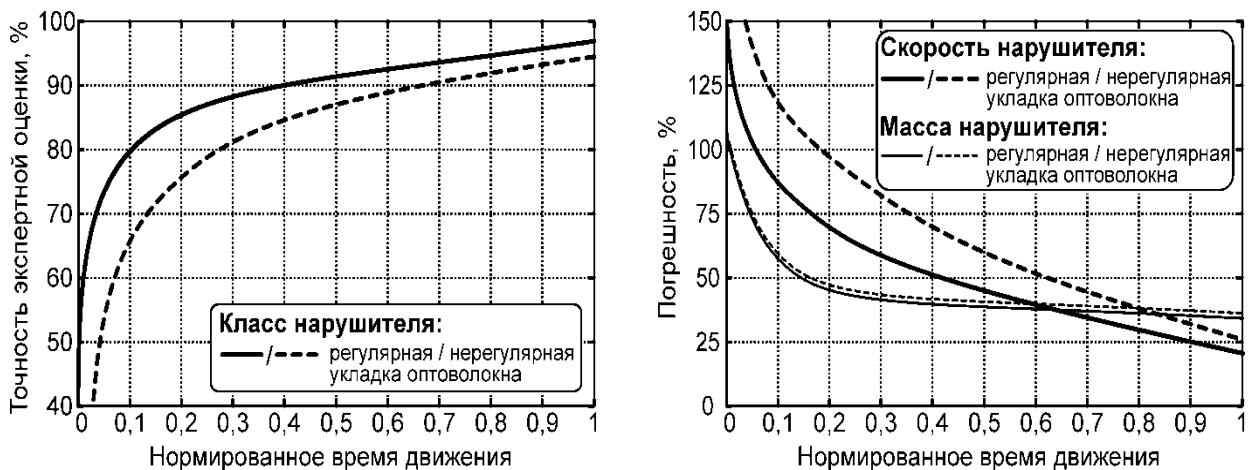


Рис. 3.8. Характеристики точности системы для различных укладок

Также была проведена серия экспериментов, которая направлена непосредственно на решение задачи локализации «участка нарушения». Исследование осуществлялось для примера «охранного периметра», содержащего 20 участков. Выполнено обучение 20-ти нейронных сетей, каждая из которых привязана к своему участку и обучалась восстанавливать 1 для него, и 0 для всех



остальных. Организация входного вектора каждой из этих нейронных сетей аналогична предыдущим экспериментам. Обучение выполнено на 10000 примерах. Результаты тестирования полученной в итоге системы оказались неудовлетворительными.

Для следующего эксперимента выполнено преобразование 80-ти элементного входного вектора нейронной сети в 10-ти элементный вектор, содержащий только информацию о последовательности порядковых номеров измерительных линий (ИЛ), на которые было оказано воздействие в процессе движения нарушителя. Разработан алгоритм, основанный на строгой логике, сравнивающий эту последовательность с фактическими последовательностями измерительных линий на каждом из 20-ти участков и генерирующий 20-ть коэффициентов подобия. В противовес этому алгоритму произведено повторное обучение 20-ти нейронных сетей, но теперь с 10-ти элементным вектором на входе.

На рисунке 3.9. представлены примеры результатов работы нейросетевого решения, а также решения, полученного на основе традиционных вычислений по алгоритму сравнения последовательностей порядковых номеров ИЛ (нарушитель периметра: «собака» на 17-м участке).

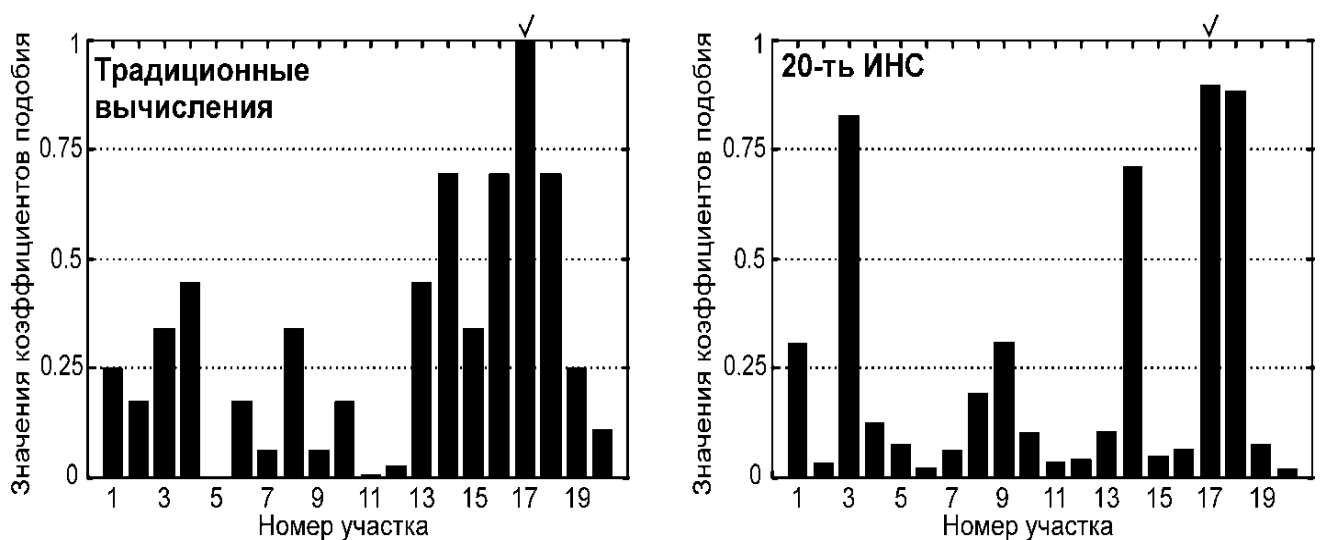


Рис. 3.9. Результаты локализации участка нарушения

Несмотря на то, что последнее нейросетевое решение по точности превзошло его предыдущий аналог, оно существенно уступает решению, полученному на базе традиционных вычислений, что можно видеть на рисунке 3.9. Этот факт подтвержден расширенным экспериментальным сравнением на примерах для «шагающих нарушителей» (участок нарушения периметра колесным транспортом, последовательно действующим на каждую измерительную линию, потенциально восстанавливается со 100% точностью, а поэтому этот тип нарушителей в данном случае не представляет интереса), где показатели точности нейросетевого решения оказались практически в два раза хуже. При этом применение нейросетевого аппарата для определения направления движения нарушителя позволило получить высокую точность: 98%.

Исследования по сгенерированным данным проводились с целью определения необходимого для выборки количества примеров измерений, получаемых от РВОИС, и допустимости сокращения размерности входного вектора нейронной сети при обработке, не вызывая при этом существенных потерь в качестве распознавания динамических образов ИРИИС. Основные результаты проведённых экспериментов с использованием многослойных нейронных сетей прямого распространения при тестировании на выборках примеров, которые не использовались при обучении, когда на вход подавали вектора характерных признаков (таблица 3.2.), представлены в таблице 3.5.

Проведенные численные эксперименты позволили выявить следующие преимущества этого способа представления входных данных:

- на вход НС достаточно подавать вектора, состоящие только из наиболее значимых признаков, а не все выделенные 13 признаков;
- выбор признаков зависит от того, какой параметр идентифицируемого объекта нужно восстановить;
- сокращается время обучения НС и работы системы при сохранении точности распознавания объекта.

Таблица 3.5. Результаты нейросетевого решения по признакам

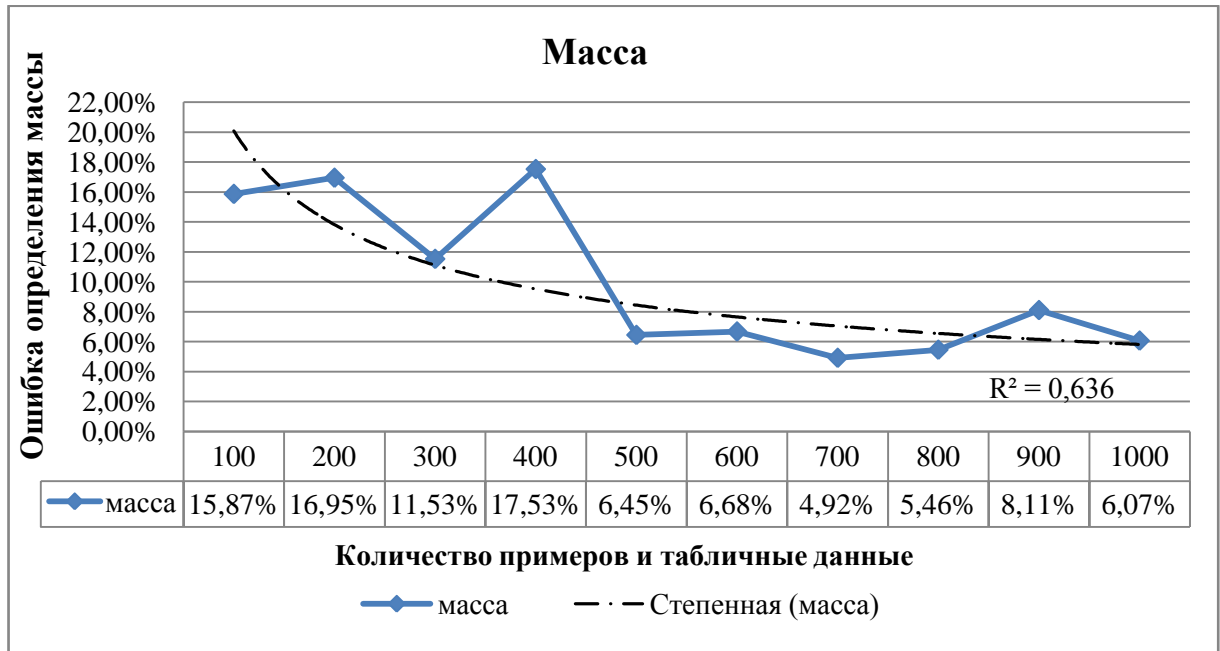
Распознаваемые параметры объекта	Значимые признаки	Ошибка, % (при использовании значимых признаков для расширенной выборки объектов)	Ошибка, % (при использовании всех признаков для ограниченной выборки объектов)		
		20000 примеров	1000 примеров	500 примеров	200 примеров
Масса, кг	1,3,5,6,13	15,16	6,07	6,45	16,95
Скорость, м/с	1,3,5,6	1,69	1,87	3,31	4,60
Класс и направление движения	3,5,6,7,8,9,11,12	6,03	2,93	4,40	5,15

При обучении и тестировании нейронной сети в выборке примеров использовали два возможных набора объектов: с ограниченным и расширенным количеством классов с учётом направлений движения. Таким образом, набор объектов с ограниченным количеством классов (или совокупность ограниченной выборки) состоял из двух классов («человек» и «авто») в сочетании с двумя вариантами поведения объекта («нарушитель» и «беглец»). Совокупность расширенной выборки была представлена набором объектов из трёх классов («человек», «авто», «фауна») в сочетании с тремя вариантами поведения объекта («нарушитель», «беглец» и «гуляющий»). Объекты сгруппированы под номерами по приоритету классов при распознавании. Ранжирование объектов по номерам помогает интерпретировать оценки критериев качества при анализе результатов распознавания.

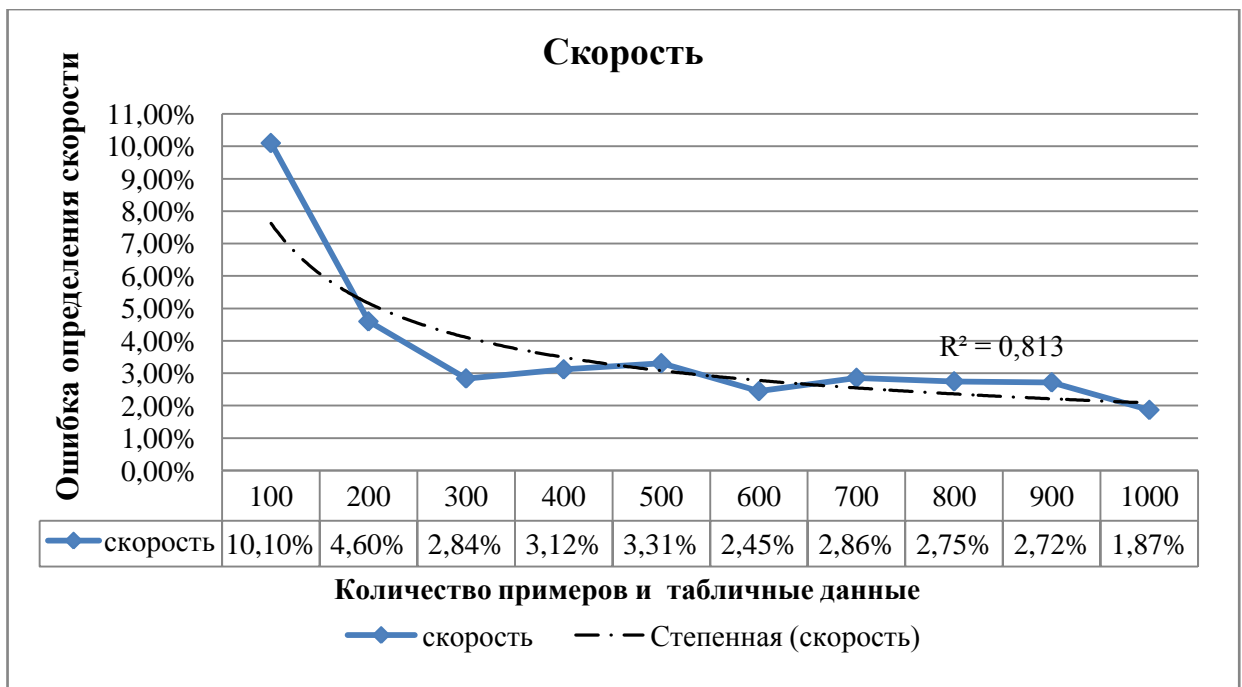
Исследование показало, что при ограниченном количестве классов для обучения нейронной сети необходимо не менее 500 примеров в выборке, а при расширенном – не менее 5000 примеров. На диаграммах (рис. 3.10 (а, б и в)) наблюдается уменьшение среднеквадратичной ошибки определения массы и скорости, а также распознавания типа (класса и направления) движущегося объекта с ростом количества примеров ограниченной выборки. При этом

количество примеров в выборке для тестирования многослойной нейронной сети прямого распространения были заданы равными. С помощью линий тренда отображаются основные тенденции изменения данных или линии скользящего среднего.

а)



б)



в)

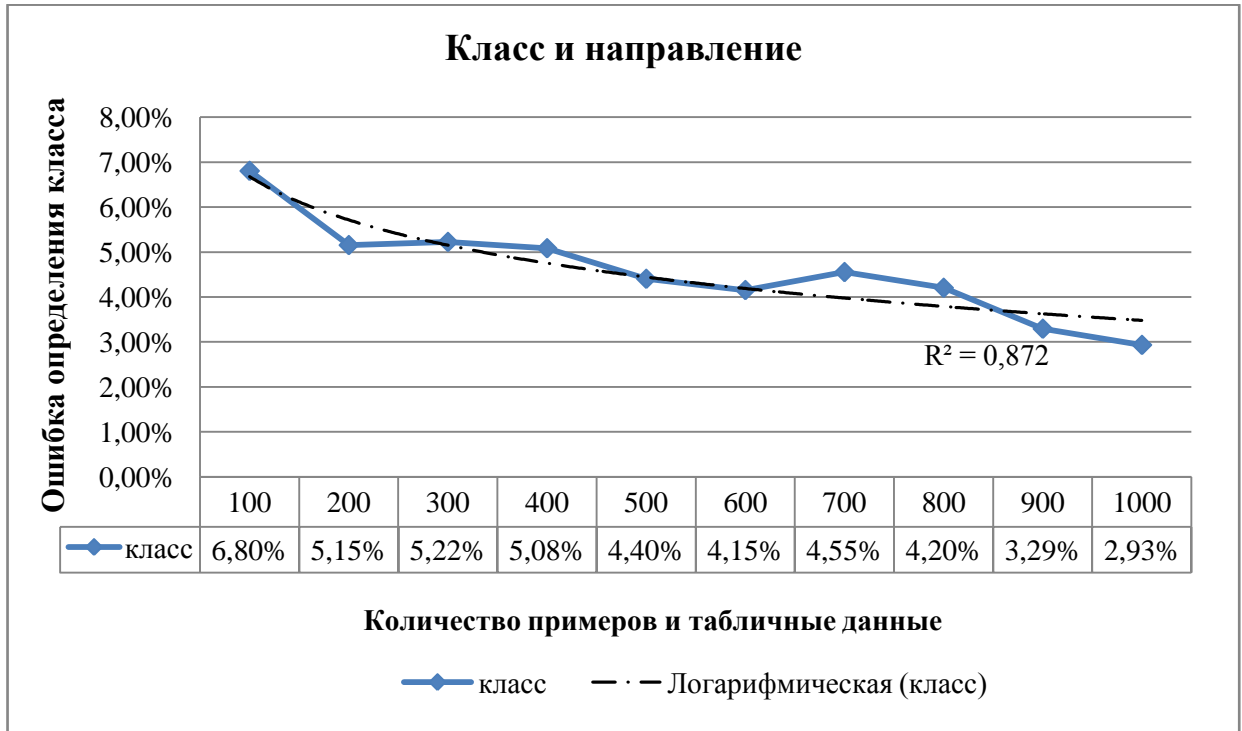


Рис. 3.10. Характеристики точности распознавания для различного количества примеров ограниченной выборки: а) масса; б) скорость; в) класс и направление

Величина достоверности аппроксимации  $R^2$  (коэффициент детерминации) отображает степень соответствия ожидаемых значений для линий тренда фактическим данным. Величина достоверности аппроксимации рассчитывалась в MS Excel автоматически. Приближение этого показателя к единице свидетельствует о хорошем совпадении расчётной линии с фактическими данными. В нашем случае величины достоверности аппроксимации получились: 0,690 для массы, 0,857 для скорости и 0,872 для класса и направления, что показывает некоторую разнородность выбранных выборок, но тенденция к уменьшению ошибки наблюдается во всех представленных диаграммах результатов экспериментов. Скорость снижения ошибки в них также уменьшается (или остаётся неизменной) и её значение постепенно стремится к 0. Таким образом, ошибка при определении массы составила 5-8%, скорости – 2-3%, а класса и направления – 3-4,5%.

Для оценки способности системы выполнять поставленную задачу распознавания движущихся объектов был выработан критерий пригодности, величина которого рассчитывалась при определении типа объекта (т. е. его класса и направления движения) подобно коэффициенту корреляции Пирсона по формуле [129]:

$$r_{xy} = \frac{n \times \sum_{i=1}^n (x_i \times y_i) - \left( \sum_{i=1}^n x_i \times \sum_{i=1}^n y_i \right)}{\sqrt{\left[ n \times \sum_{i=1}^n x_i^2 - \left( \sum_{i=1}^n x_i \right)^2 \right] \times \left[ n \times \sum_{i=1}^n y_i^2 - \left( \sum_{i=1}^n y_i \right)^2 \right]}}, \quad (3.1.10.)$$

$x_i$  – значения независимой переменной (фактический номер типа объекта);  $y_i$  – значения зависимой переменной (номер объекта, полученный после распознавания его класса и направления нейронной сетью по признакам);  $n$  – количество наблюдений (примеров).

Критерий пригодности составил около 0,8 для ограниченной выборки (рис. 3.11.), что с учётом характеристик точности свидетельствует о возможности решения поставленной задачи предложенной в работе системой.

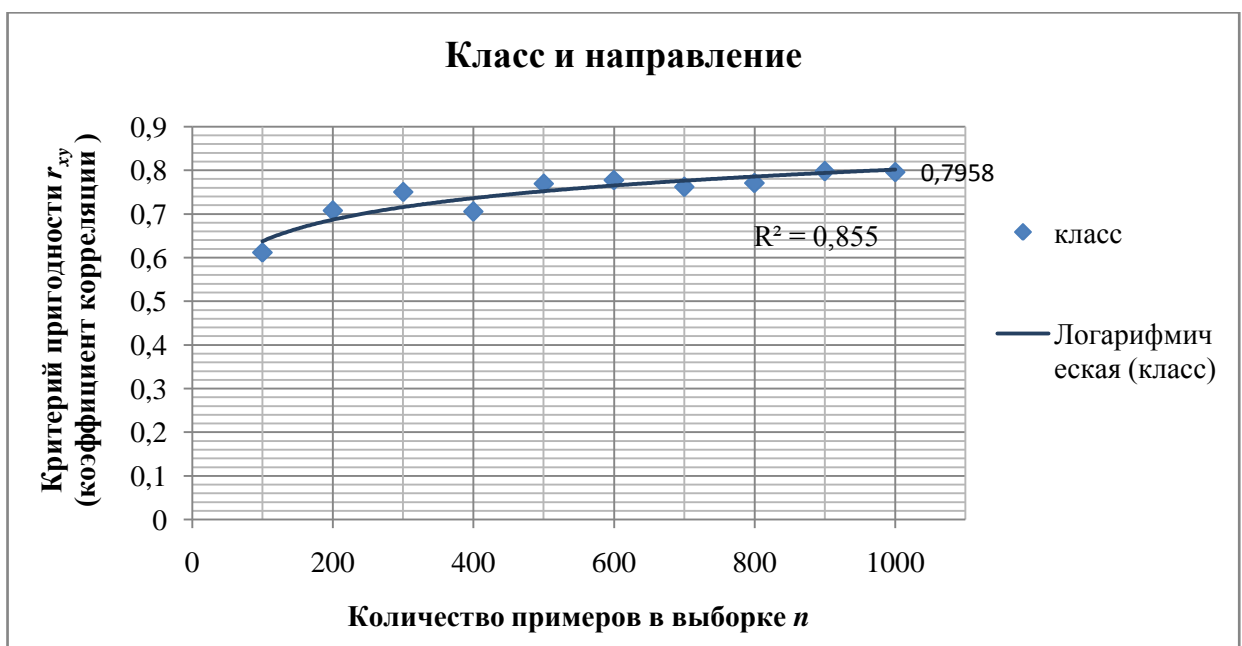
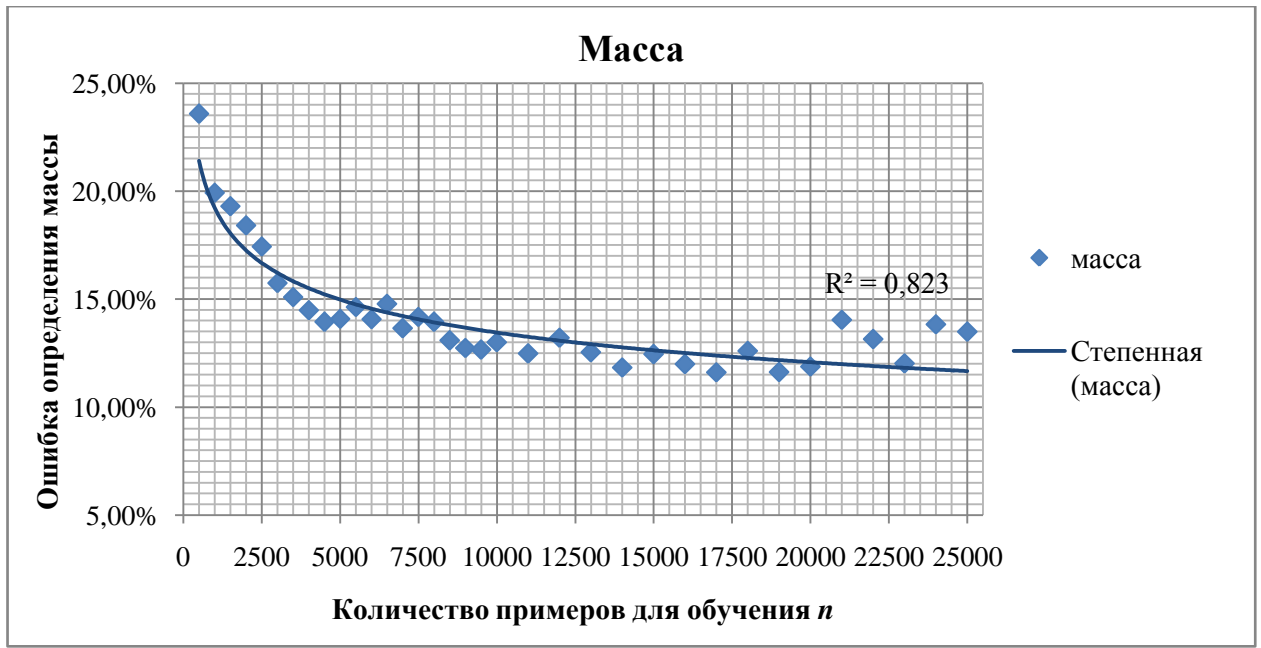


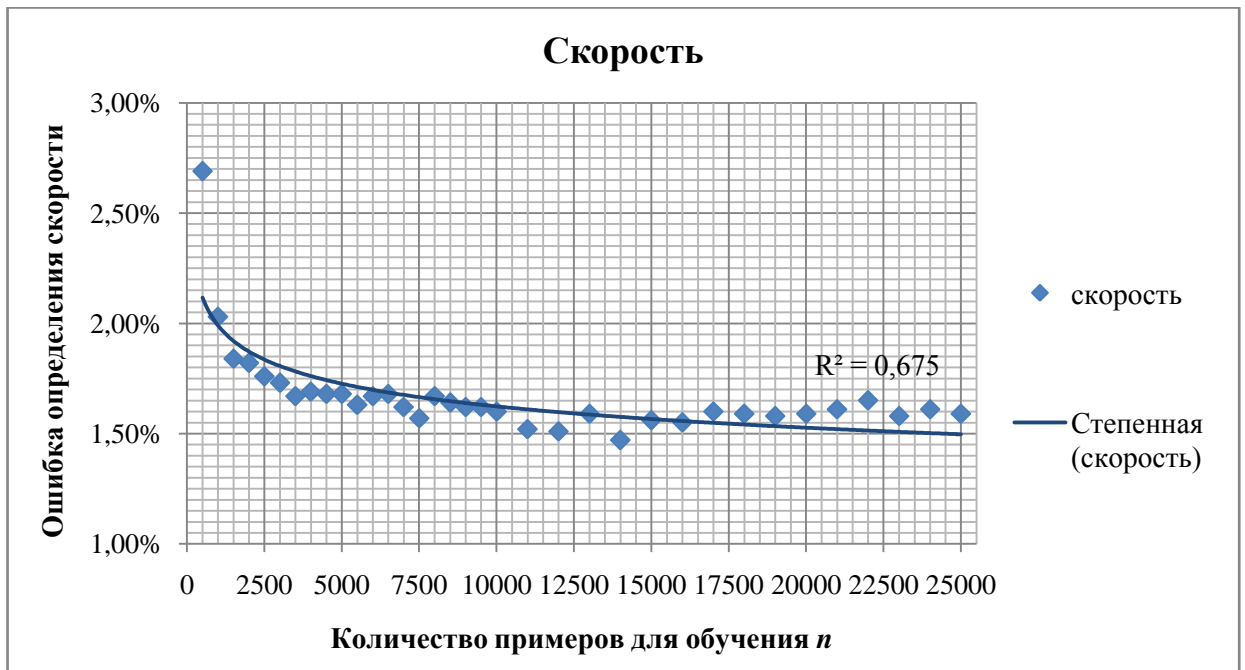
Рис. 3.11. Критерий пригодности системы для выполнения поставленной задачи для различного количества примеров ограниченной выборки

На рисунке 3.12 (а, б и в) по графикам можно видеть, что ошибка для расширенной выборки при определении массы составила 13-15%, скорости – около 1,6%, класса и направления – 5-6%.

а)



б)



в)

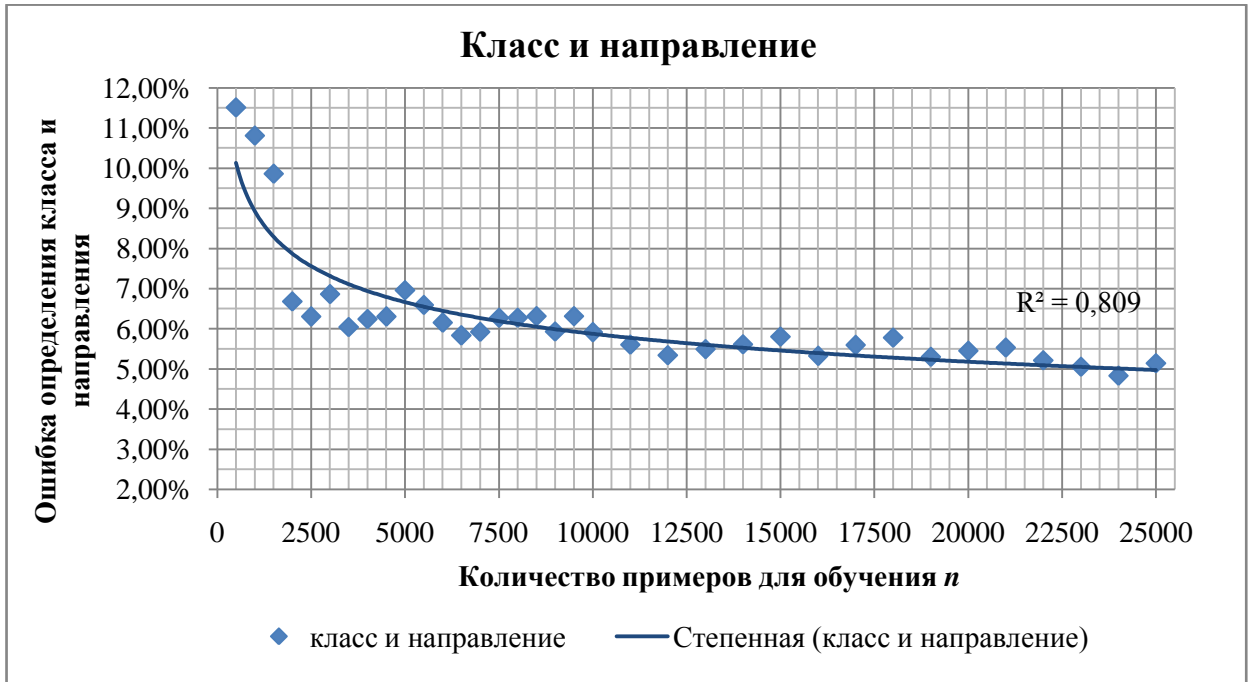


Рис. 3.12. Характеристики точности распознавания для различного количества примеров расширенной выборки: а) масса; б) скорость; в) класс и направление

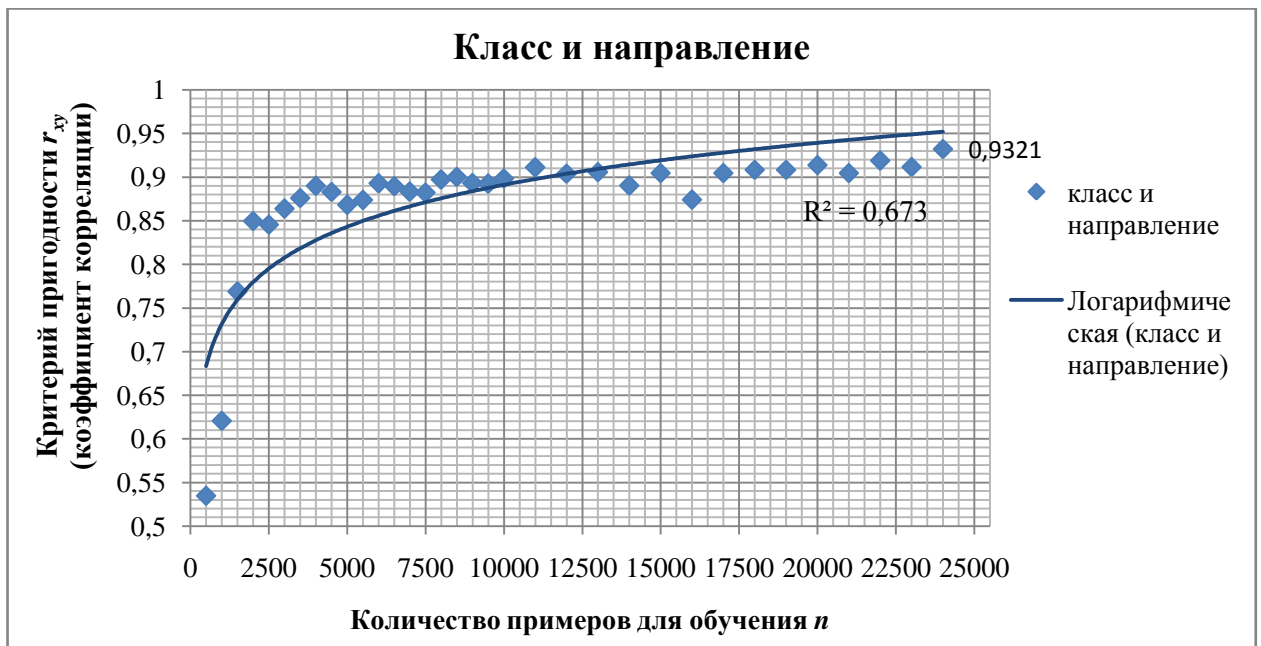


Рис. 3.13. Критерий пригодности системы для выполнения поставленной задачи для различного количества примеров расширенной выборки

Для расширенной выборки полученные результаты при оценке критериев качества работы системы имеют аналогичные тенденции к изменению, что и при



ограниченной выборке примеров, а значение критерия пригодности достигает 0,9-0,95 с той лишь разницей, что для поддержания репрезентативности выборки количество примеров существенно больше (рис. 3.13).

Величина достоверности  $R^2$  принимает значения 0,675, 0,823 и 0,809 для скорости, массы и типа (класса и направления) объекта, соответственно. Время обучения нейронной сети линейно возрастает с ростом объёма выборки, как показано на рисунке 3.14.

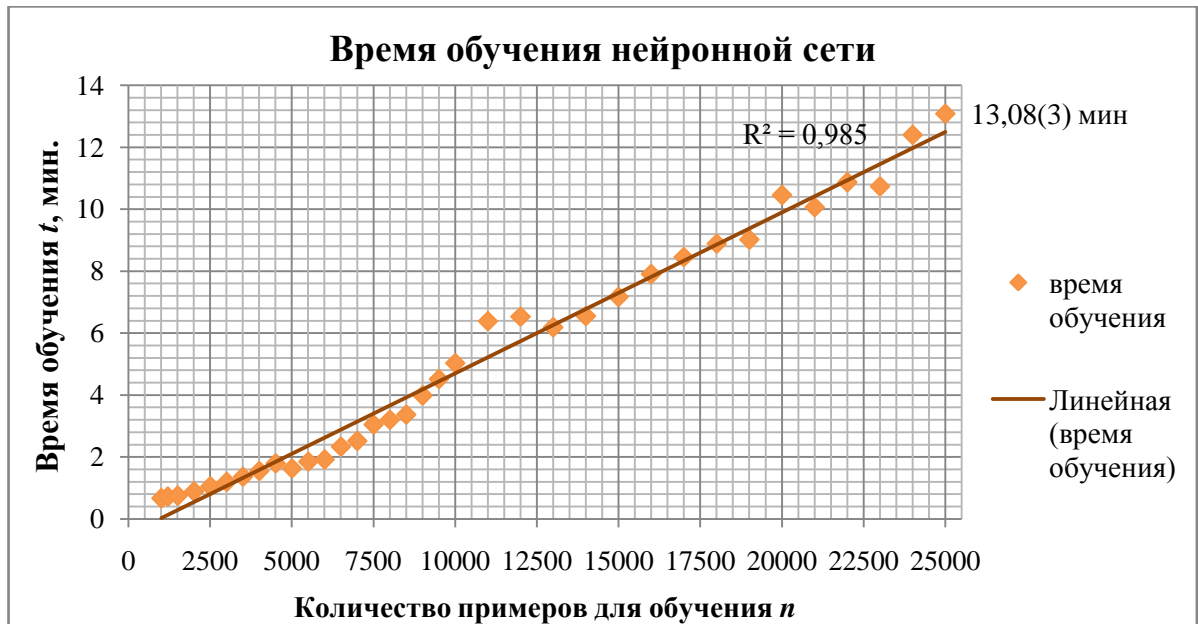


Рис. 3.14. Зависимость времени обучения от количества примеров в выборке

Для подтверждения предположения, что количество задаваемых признаков может быть уменьшено без существенного понижения качества распознавания, были проведены численные эксперименты и выделены наиболее значимые признаки. Отбор значимых признаков осуществлялся методом направленного перебора (метод ветвей и границ) [158]. Этот метод позволяет отсеять неперспективные планы (наборы сочетаний признаков) и перейти от полного перебора планов к сокращённому перебору. Общая постановка такой задачи имеет вид:

$$f(X) \rightarrow \min, \quad X \in P, P - \text{множество всех признаков,}$$

$f(X)$  – целевая функция, скалярная функция своих аргументов.

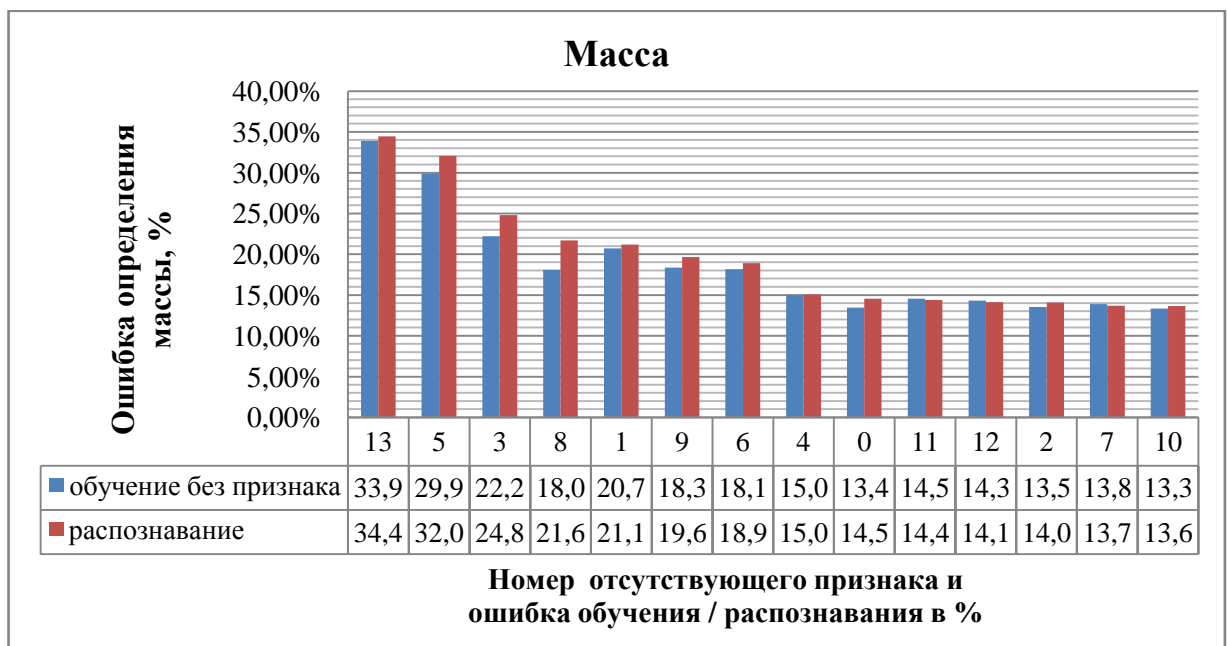
Основные этапы алгоритма отбора значимых признаков:

1. Ранжирование признаков в порядке уменьшения ошибки распознавания системой параметров (масса, скорость) и типа (класс и направление) движущегося объекта, которая возникает в случае исключения одного признака из входного вектора нейронной сети (рис. 3.15 (а, б и в)). Критерий пригодности системы, показанный на рис. 3.16., демонстрирует разную чувствительность системы на отсутствие отдельного признака при распознавании.

В начале, после ранжирования оставляем  $n$  признаков, ошибка определения параметров, при отсутствии которых, выше или равна той, что была получена при распознавании по всем признакам.

На каждом  $i$ -ом ( $i=0,1,2,\dots,(n-1)$ ) шаге количество признаков в подмножестве уменьшается на один условно «слабый» признак и остаётся  $(n-i)$  признаков, из которых формируются подмножества  $P_i$ , где все признаки, кроме последнего, оставляют неизменными, а последний перебором меняют на ближайшие сокращённые признаки на  $(i-1)$ -ом шаге.

а)



б)



в)

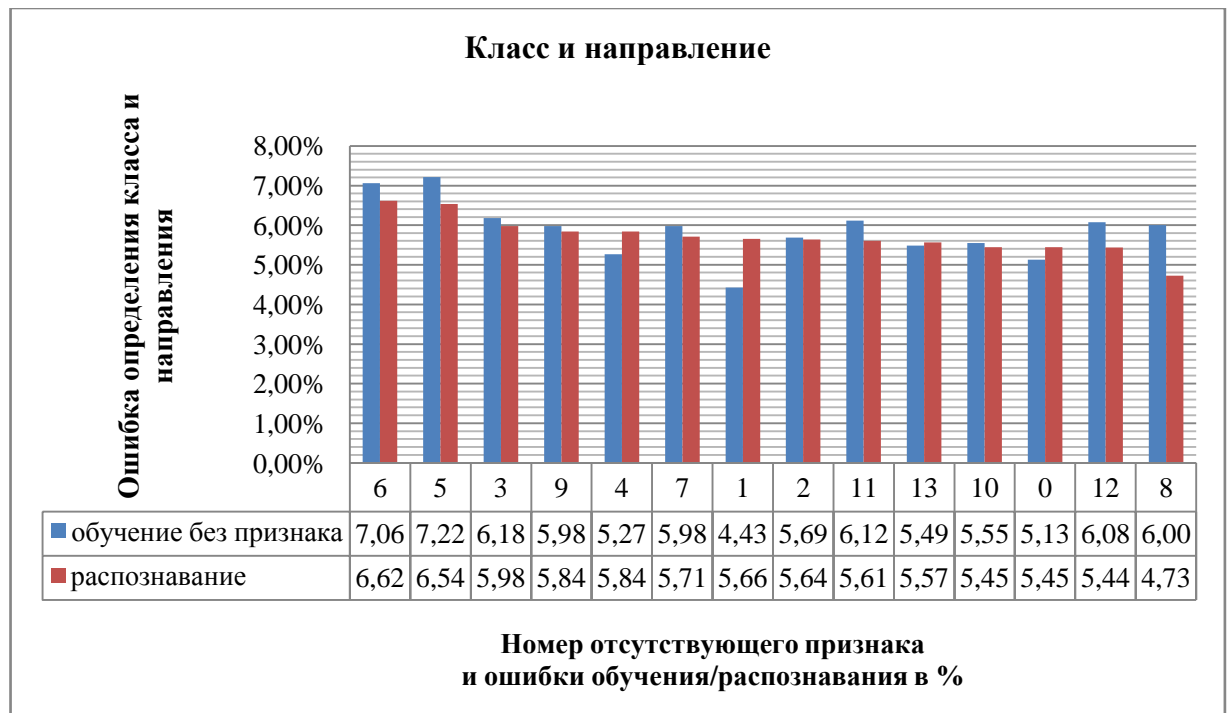


Рис. 3.15. Ранжирование признаков по ошибке определения: а) массы; б) скорости; в) класса и направления

В итоге оставляют тот набор признаков, при котором ошибка определяемого параметра движущегося объекта является наименьшей на  $i$ -

ом шаге. Если выбор наименьшей ошибки затруднён, так как ошибки оказываются примерно равными, с точностью до 0,1% для разных подборов последних признаков в наборе признаков, то считают, что это альтернативные наборы, и на следующем шаге учитывают последние признаки всех этих наборов при переборе. Затем, вычисляется нижняя граница возрастания ошибки параметра на  $i$ -ом шаге  $f(X_i)$  и сравнивается их

сумма за  $i$  шагов с нижней границей целевой функции:  $\sum_{k=0}^i f(X_k) \leq f(X)$ .

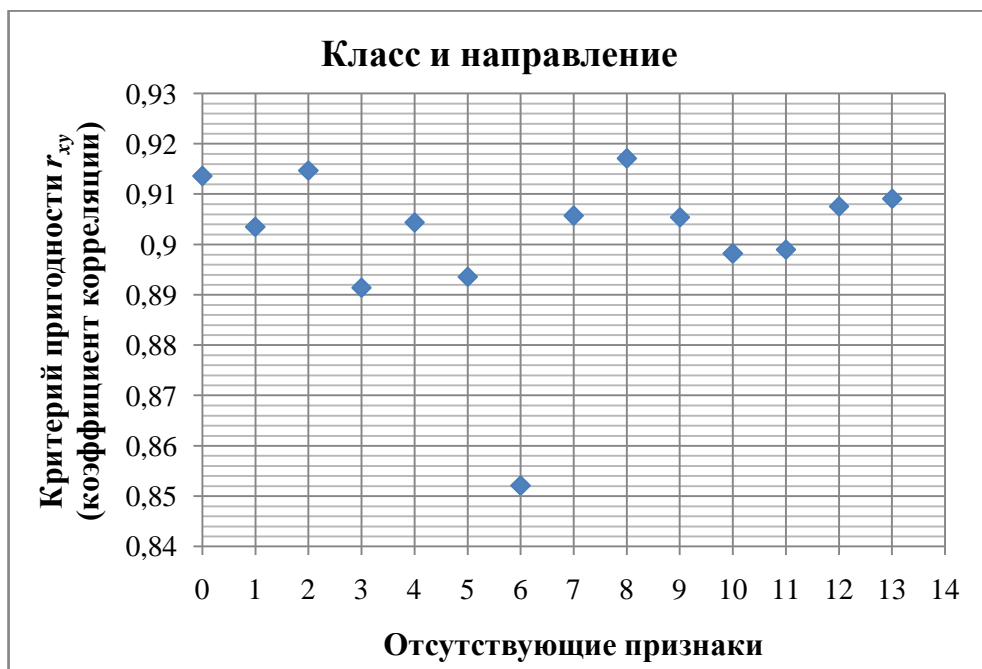


Рис. 3.16. Критерий пригодности системы при отсутствии одного из признаков

2. При достижении целевой функцией заданного  $\min$  значения для нижней границы фиксируется набор значимых признаков или альтернативные наборы.
3. Для полного анализа, следующие шаги отбора признаков выполняют также как и раньше, но при этом находят превышение суммы для нижних границ возрастания ошибки определения параметра за  $i$  шагов над нижней

границей целевой функции, то есть:  $\sum_{k=0}^i f(X_k) - \min > 0$ .

Результаты использования данного алгоритма позволили установить, что для выполнения задачи достаточно использовать не весь набор выделенных признаков, а набор значимых признаков, которые указаны в таблице 3.5. Исследования показали, что критерий пригодности системы для выполнения поставленной задачи достигает 0,91 (рис. 3.17) и зависит от количества признаков. При этом критерии пригодности между собой отличаются незначительно (в пределах  $\pm 0,02$ ) при использовании 8 признаков и более для идентификации движущегося объекта.

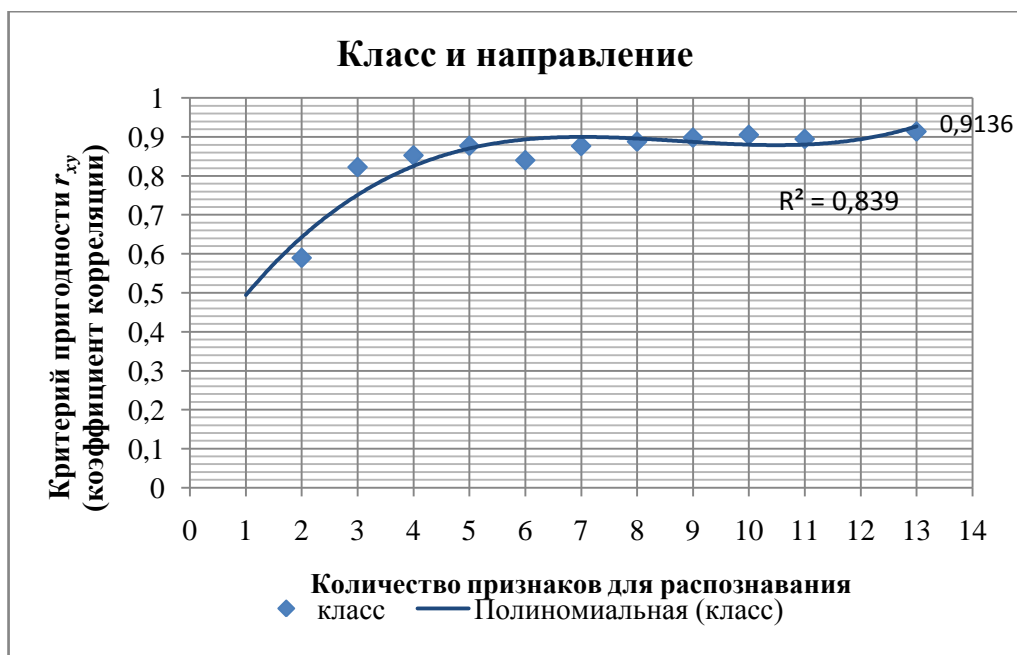


Рис. 3.17. Критерий пригодности системы при разном количестве признаков

Характер изменения ошибок определения параметров (массы, скорости, класса и направления) объекта можно видеть на рисунках 3.18. и 3.19(а, б, в).

В работе выявлено, что для каждого определяемого параметра (рис. 3.18. и 3.19 (а, б и в)) при распознавании существует оптимальное минимальное количество признаков, при котором ошибка параметра отличается незначительно ( $\leq 1\%$ ) от её значения, которое получают при использовании полного набора выделенных признаков.

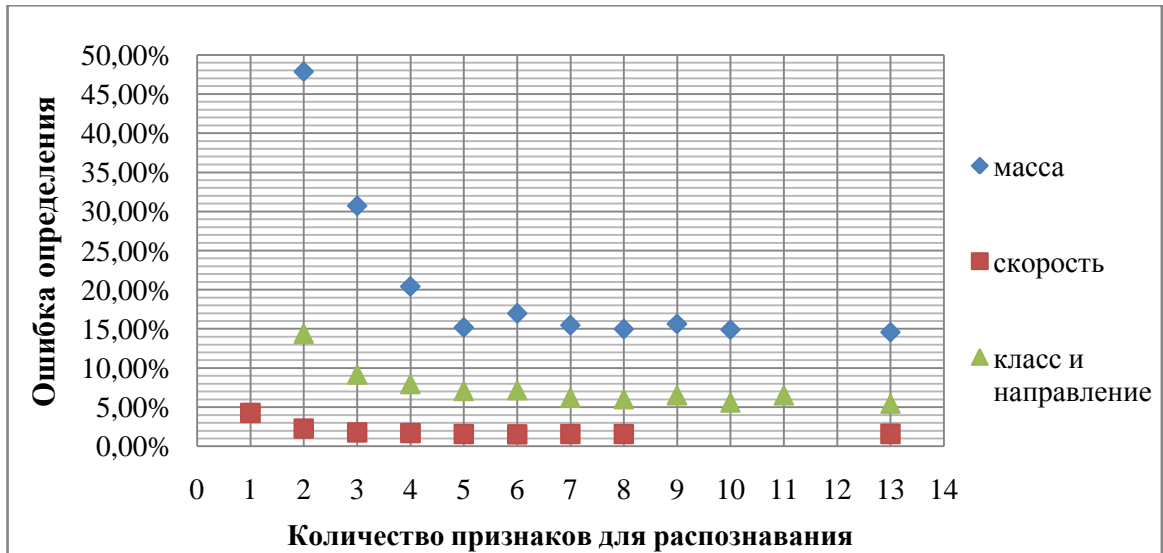
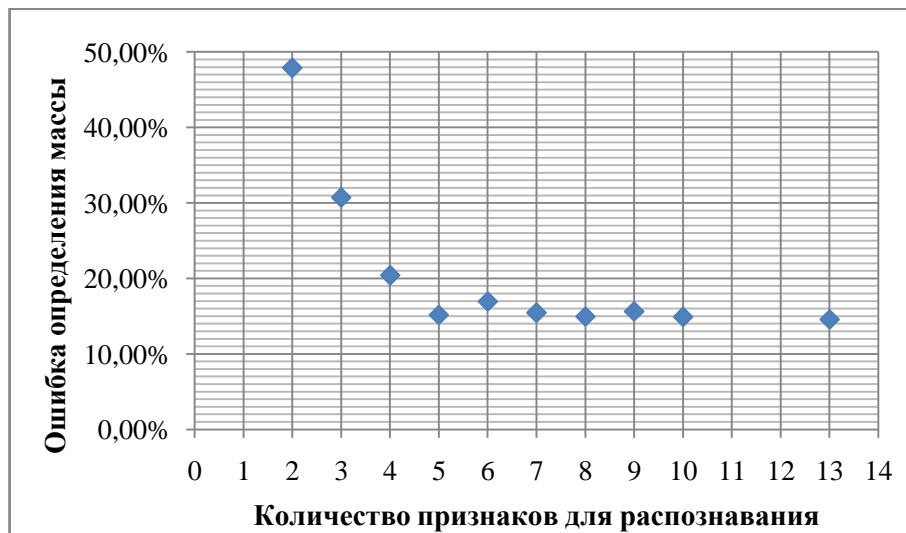
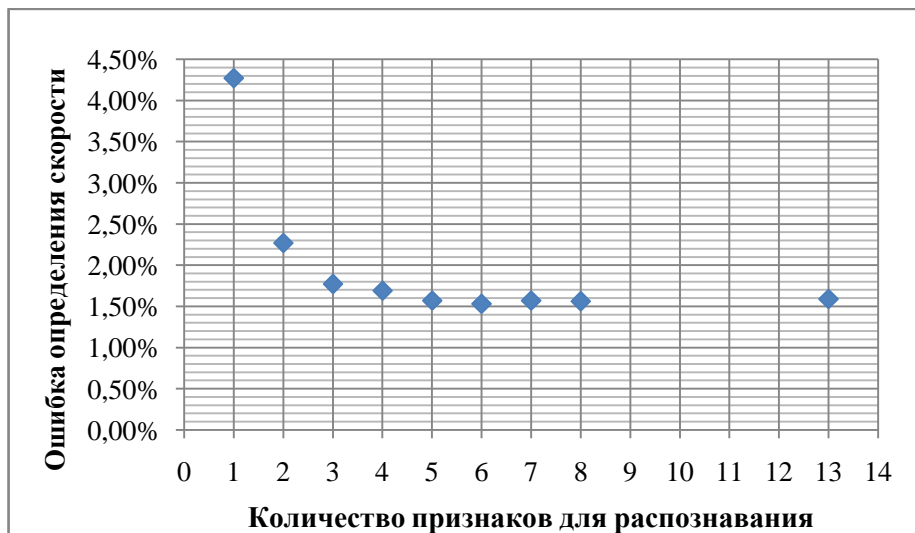


Рис. 3.18. Ошибки определения параметров объекта в сравнении

а)



б)



в)

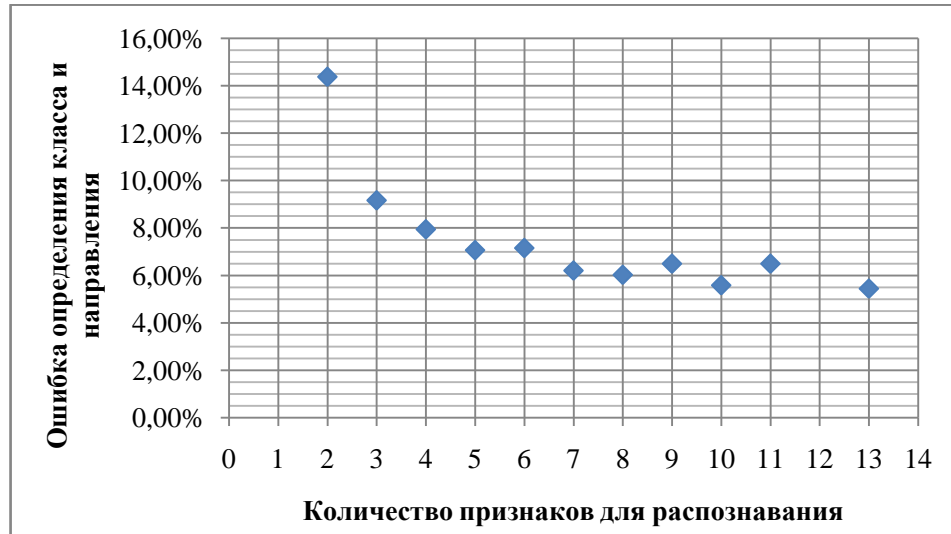


Рис. 3.19. Ошибки определения: а) массы; б) скорости; в) класса и направления

Таким образом, проведённая серия экспериментов подтвердила преимущества предлагаемого способа, основанного на формировании МО по признакам, и способность разработанной системы выполнять распознавание динамических образов и определять параметры движущихся объектов лишь по значимым характерным признакам, полученным из сгенерированных данных, при использовании в обработке многослойных нейронных сетей прямого распространения.

### 3.1.6 Результаты исследований при обработке реальных данных с макета ВОИС охраны периметра с применением нейронных сетей

Процедуры для вычисления главных компонент и выделения характерных признаков были описаны выше в разделах 3.1.3 и 3.1.4. Поэтому в данном разделе главы подробнее остановимся на самих результатах обработки многослойными нейронными сетями прямого распространения по МО, сформированным одним из двух способов (методом главных компонент или выделением характерных признаков), для реальных данных с макета ВОИС охраны периметра [176, 179, 180]. Основной целью этого этапа исследований было показать пригодность и эффективность предлагаемых способов для

обработки реальных данных, сделать оценку и сравнить возможности системы при распознавании параметров реального движущегося объекта (массы, длины пути, класса и направления движения) по 20 признакам и по 19 главным компонентам с учётом времени инцидента. Для численного исследования были созданы программы (приложение №3) в среде MATLAB, позволяющие выполнить поставленную задачу. Результаты обработки реальных данных многослойными нейронными сетями прямого распространения представлены в таблице 3.6.

Таблица 3.6. Результаты обработки реальных данных нейронными сетями

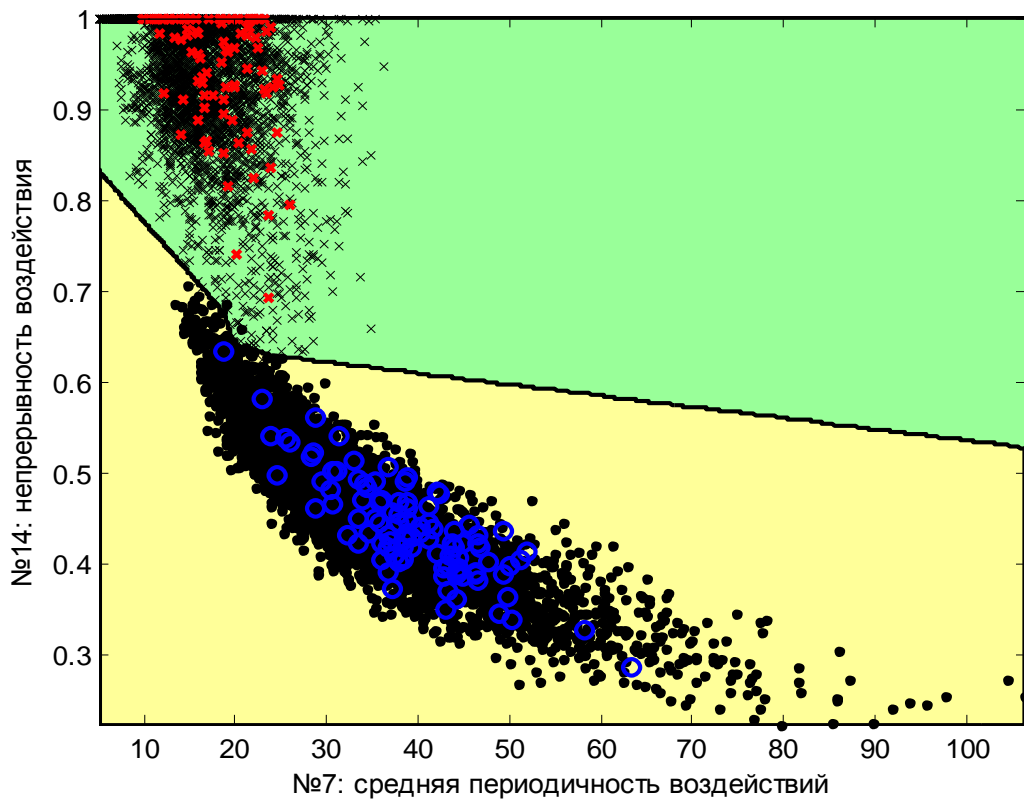
Параметр движущегося объекта	Ошибка по обучающей выборке (10000 примеров), %		Ошибка по тестовой выборке (200 примеров), %		Значимые признаки	Количество главных компонент
	По признакам	По главным компонентам	По признакам	По главным компонентам		
Класс	0,04	0,01	0	0	7,14	2
Направление движения	0	0	0	0	19,20	4
Масса	11,7967	14,3098	12,9418	11,9496	5,14	9
	10,7624		11,0035		3,5,15	
	9,9818		10,3132		3,5,6,15	
	<u>10,2127</u>		<u>9,7757</u>		3,5,6,15, 20	
	10,1200		10,6965		1,5,6,11, 14,15	
Длина пути	8,4706	7,5857	8,1024	7,7334	3,15	19
	8,1897		8,1161		3,5,15	
	7,4447		7,7145		3,6,15,19	
	6,8433		7,0570		3,5,6,15, 19	
	<u>6,6221</u>		<u>6,5462</u>		3,4,5,6,15,19	
	6,7787		6,8614		3,5,6,8,10,15,20	
	6,7539		6,9505		3,5,6,8,10,12,15, 20	



По таблице 3.6. можно видеть, что класс и направление движения распознаются с ошибкой 0-0,01%, т.е. практически безошибочно. При определении массы точность ожидаемо ниже и составляет  $\approx 10\%$  по пяти признакам и  $\approx 12\%$  по девяти главным компонентам, а при нахождении длины пути она оказалась  $\approx 6,5\%$  по шести признакам и  $\approx 8\%$  по 19 главным компонентам. Результаты показали, что количество требуемых признаков для распознавания можно ограничить, выделив значимые из них, и оно значительно ниже, чем количество необходимых главных компонент. В таблице 3.6. выделены наиболее удачные наборы признаков для определения массы и длины пути.

Картина распределения примеров для класса и направления движения на рисунке 3.20 (а, б) наглядно демонстрирует хорошее разделение объектов всего лишь по двум признакам.

а)



б)

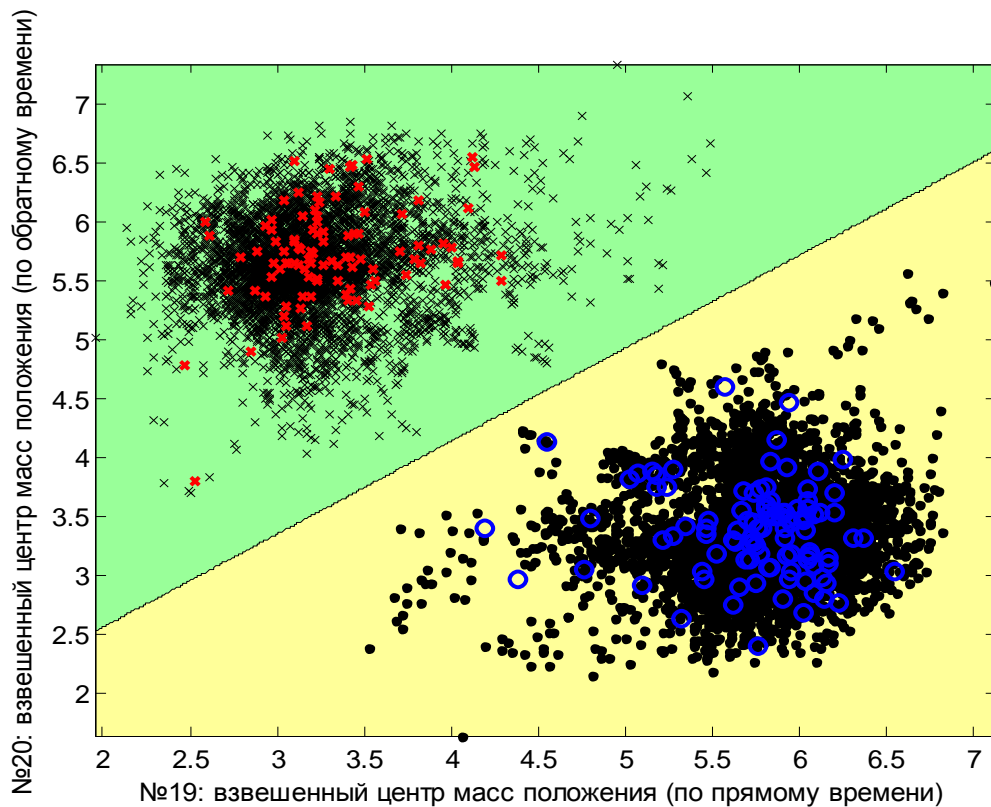
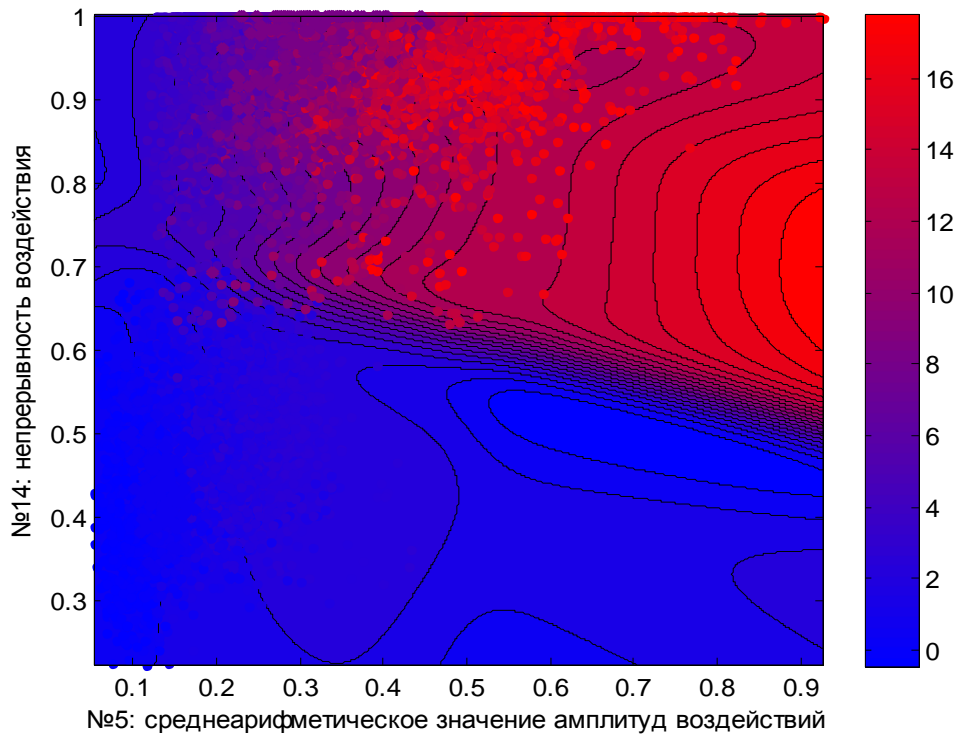


Рис. 3.20. Картина распределения примеров обучающей и тестовой выборок по двум признакам при распознавании: а) класса; б) направления движения

По рисунку 3.21 (а, б) видно, что картины распределения примеров по двум признакам для массы и длины пути не имеют столь четко очерченных границ разделения, но, тем не менее, система способна с приемлемой точностью определять эти параметры объекта.

Таким образом, в ходе проведения численных экспериментов проверена целесообразность и показана эффективность предлагаемых способов обработки данных, как по сгенерированным данным, так и по реальным данным, полученным от макета РВОИС охраны периметра.

а)



б)

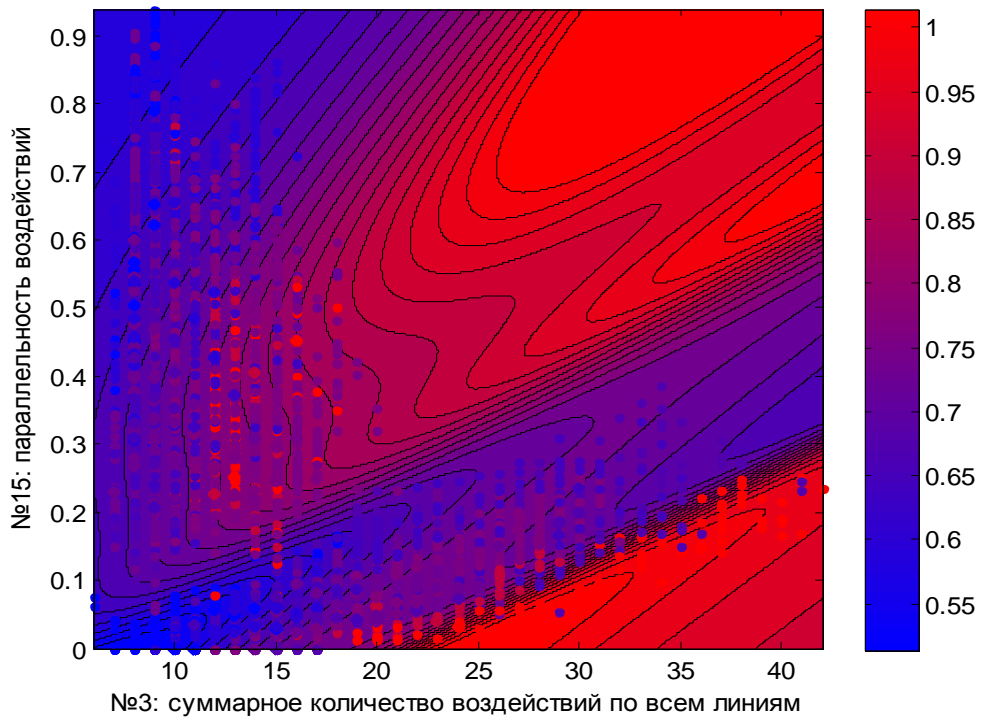


Рис. 3.21. Картина распределения примеров обучающей и тестовой выборки по двум признакам при распознавании: а) массы, кг; б) длины пути, м.

### 3.2 ОБРАБОТКА ДАННЫХ ДЛЯ РАСПОЗНАВАНИЯ ДИНАМИЧЕСКОГО ОБРАЗА С ИСПОЛЬЗОВАНИЕМ ПРОДУКЦИОННОЙ НЕЧЁТКОЙ СИСТЕМЫ

Продукционные системы были разработаны в рамках исследований по методам искусственного интеллекта и нашли широкое применение для представления знаний и вывода заключений в экспертных системах, основанных на правилах продукций [63-65, 158-162]. Поскольку нечёткий вывод реализуется на основе нечётких продукционных правил, рассмотрение базового формализма нечётких продукционных моделей приобретает самостоятельное значение. При этом нечёткие правила продукций не только во многом близки к логическим моделям, но и, что наиболее важно позволяют адекватно представлять практические знания экспертов в той или иной области. Использование нечёткой логики в алгоритмах получения нечётких заключений на основе нечётких условий предназначено для реализации процесса нечёткого вывода в продукционных нечётких системах. Такие системы позволяют решать задачи автоматического управления, классификации данных, распознавания образов, принятия решений и многие другие [66-68, 163-174].

Правило нечёткой продукции [167-172]. В общем случае под *правилом нечёткой продукции* или просто – *нечёткой продукцией* принимается выражение следующего вида:

$$(i): Q; P; \mathcal{A} \Rightarrow \mathcal{B}; S, F, N, \quad (3.2.1)$$

где  $(i)$  – имя нечёткой продукции;  $Q$  – сфера применения нечёткой продукции,  $P$  – условие применимости ядра нечёткой продукции,  $\mathcal{A} \Rightarrow \mathcal{B}$  – ядро нечёткой продукции, в котором  $\mathcal{A}$  – условие ядра (или антецедент);  $\mathcal{B}$  – заключение ядра (или консеквент); " $\Rightarrow$ " – знак логической секвенции (или следования);  $S$  – метод или способ определения количественного значения степени истинности заключения  $\mathcal{B}$  на основе известного значения степени истинности условия  $\mathcal{A}$ . Данный способ в общем случае определяет так называемую *схему* или алгоритм

нечёткого вывода и называется также *методом композиции* или *методом активации* согласно Стандарту ИЕС 1131-7;  $F$  – коэффициент определённости или уверенности выражает количественную оценку степени истинности или относительный вес нечёткой продукции. Коэффициент уверенности принимает своё значение из интервала  $[0,1]$  и часто называется *весовым коэффициентом* нечёткого правила продукции;  $N$  – *постусловия* продукции. В качестве  $A$  и  $B$  могут использоваться составные логические нечёткие высказывания, т.е. элементарные нечёткие высказывания, соединённые нечёткими логическими связками, такими как нечёткое отрицание, нечёткая конъюнкция и нечёткая дизъюнкция.

Продукционная нечёткая система [168-173]. *Продукционная нечёткая система* или *система нечётких правил продукции* представляет собой некоторое согласованное множество отдельных *нечётких продукций* или правил нечётких продукций в форме «Если  $A$ , то  $B$ » (или в виде: «IF  $A$ , THEN  $B$ »), как определено в Стандарте ИЕС 1131-7).

Основные этапы нечёткого вывода в продукционной нечёткой системе следующие:

- Формирование базы правил систем нечёткого вывода;
- Фаззификация входных переменных;
- Агрегирование подусловий;
- Активизация подзаключений;
- Аккумуляирование заключений;
- Дефаззификация.

Алгоритмы для формирования базы нечётких правил и реализации нечёткого вывода заключений на основе условий в рамках решения задачи распознавания динамических образов могут быть использованы при обработке сигналов, получаемых от распределённых волоконно-оптических измерительных сетей. Разработка таких алгоритмов позволяет ускорить процесс обработки данных РВОИС, что делает возможным работу системы в режиме реального

времени. Автоматизация процесса построения системы, обеспечение адаптивности и повышение эффективности её работы, не требующее высокой квалификации экспертов при эксплуатации, – это ряд преимуществ, которые даёт реализация подобных алгоритмов в системе и поэтому они могут иметь особое прикладное значение при создании интеллектуальных информационно-измерительных систем и современных систем поддержки принятия решений.

Целью данного исследования явилась разработка и анализ алгоритма построения продукционной нечёткой системы в виде нечёткого дерева решений применительно к ВОИС охраны периметра объекта, процесс обучения которой основан на использовании множественной выборки экспериментальных данных.

### **3.2.1 Реализация продукционной нечёткой системы в виде нечёткого дерева решений по реальным данным с макета ВОИС охраны периметра**

Основные этапы нечёткого вывода в алгоритме построения нечёткого дерева решений осуществляют посредством выполнения двух режимов программы: обучения и тестирования [176-178]. В режиме обучения используют 10000 примеров, полученных с помощью модуля преобразования измеренных данных с макета ВОИС охраны периметра (см. раздел 3.1.3.). Формирование базы нечётких правил основано на пошаговом режиме обучения, в результате которого получают нечёткое дерево решений. В каждом узле дерева при обучении корректируют значения и интерполируют функции принадлежности с целью последующей фаззификации входных переменных (признаков или главных компонент) при этом используют метод направленного перебора. На этапах агрегирования подусловий и активизации подзаключений получаем оптимизацию количества нечётких логических правил, необходимых значимых признаков или главных компонент. Аккумуляция заключений и дефаззификация происходят также пошагово, в каждом узле дерева решений. На каждом шаге получаем ошибку распознавания динамических образов объектов в виде соотношения количества объектов с ошибочно определёнными параметрами к общему их количеству, что служит критерием качества обучения системы. Для тестирования

построенного нечёткого дерева решений используют 200 примеров реальных данных, полученных при использовании макета ВОИС. Блок-схема алгоритма построения нечёткого дерева решения и программа, реализующая этот алгоритм для нечеткой продукционной системы в среде MATLAB, представлены в приложениях №4 и №5. Обучение происходит циклично, каждый цикл формирует один узел нечёткого дерева решений и при этом происходит отбор тех примеров из выборки, для которых полученный вид функции принадлежности узла позволяет прийти к результату распознавания, а оставшиеся примеры продолжают участвовать в процессе определения функций принадлежности следующих узлов. Далее проверяют критерий качества обучения системы, сравнивая его с целевым значением. При достижении целевого значения считают, что построение нечёткого дерева решений завершено и создана база нечётких правил.

По структуре алгоритм включает традиционные этапы нечёткого вывода продукционной нечёткой системы, но так как неизвестны нечёткие правила и функции принадлежности, то выделение этих правил и определение функций принадлежности, по сути, является главной частью содержания программы. Рассмотрим подробнее некоторые важные аспекты реализации основных этапов предложенного алгоритма:

1. На этапе *фаззификации*, при поиске функции принадлежности, они задаются как функции кусочно-линейного типа. Для каждого признака или главной компоненты множества  $A = \{a_1, a_2, \dots, a_m\}$  для 10000 примеров, полученных после предварительной обработки данных, постепенно изменяя с шагом 0,05 в интервале  $[0,1]$ , перебирают возможные значения функции принадлежности  $\mu(x)$  по оси  $OY$  в каждой выбранной точке  $x \in A$  на оси  $OX$ , где отложены не нечёткие значения.

Эта процедура выполняется до тех пор, пока снижение значения целевой функции (ошибка распознавания объектов по параметрам) не станет меньше 0,25%. При достижении этого порога фиксируют все

значения искомой функции принадлежности и производят её интерполяцию на 20 выделенных интервалах. Этап фаззификации считается законченным, когда будут найдены все значения  $b_i' = \mu(a_i)$ . Множество значений  $b_i'$  обозначим через  $\mathbf{B} = \{b_i'\}$ .

2. Этап *агрегирования* служит для определения степени истинности условий по каждому из возможных правил системы нечёткого вывода. Если условие правила состоит из одного подусловия, то степень его истинности равна соответствующему значению  $b_i'$ . Предполагая, что условие состоит из нескольких подусловий вида « $\beta_1$  есть  $\alpha'$ » ОП « $\beta_2$  есть  $\alpha''$ », где ОП – некоторая из бинарных операций нечёткой конъюнкции "И" или нечёткой дизъюнкции "ИЛИ", а  $\beta_1$  и  $\beta_2$  - различные лингвистические переменные, определяется степень истинности сложного высказывания на основе известных значений истинности подусловий  $b_i'$ . При этом для определения результата нечёткой конъюнкции или связки "И" и для нечёткой дизъюнкции или связки "ИЛИ" была использована одна из ниже представленных формул:

$$\begin{aligned} T(\mathcal{A} \wedge \mathcal{B}) &= \min\{T(\mathcal{A}), T(\mathcal{B})\} \text{ или } T(\mathcal{A} \wedge \mathcal{B}) = T(\mathcal{A}) \cdot T(\mathcal{B}) \text{ (для связки "И");} \\ T(\mathcal{A} \vee \mathcal{B}) &= \max\{T(\mathcal{A}), T(\mathcal{B})\} \text{ или } T(\mathcal{A} \vee \mathcal{B}) = T(\mathcal{A}) + T(\mathcal{B}) - T(\mathcal{A}) \cdot T(\mathcal{B}) \\ &\text{(для связки "ИЛИ").} \end{aligned} \quad (3.2.2)$$

В качестве аргументов соответствующих логических операций используют  $b_i'$ . Тем самым находят количественные значения истинности для всех условий правил системы нечёткого вывода. Оптимизируем полученные условия и задаём бинарную операцию "И" или "ИЛИ" для них по наименьшему полученному значению целевой функции. Этап агрегирования считается законченным, когда найдены все значения  $b_k''$  для каждого из правил  $R_k$ , входящих в рассматриваемую базу правил  $\mathbf{P}$  системы нечёткого вывода. Это множество значений обозначим через  $\mathbf{B}'' = \{b_1'', b_2'', \dots, b_n''\}$ .



3. До начала этапа *активизации* известны и оптимизированы значения истинности всех условий системы нечёткого вывода, то есть множество значений  $\mathbf{B}''=\{b_1'', b_2'', \dots, b_n''\}$  и найдены значения весовых коэффициентов  $F_i$  для каждого правила. На основе полученных результатов выделены значимые признаки или главные компоненты, которые и определяют выводы нечёткого дерева решений для данного его узла. Далее рассматривается каждое из заключений правил системы нечёткого вывода.

- Если в правиле только одно подзаключение (искомый параметр объекта), то степень истинности заключения равна алгебраическому произведению соответствующего значения  $b_i''$  на весовой коэффициент  $F_i$ .
- Если же заключение состоит из нескольких подзаключений, причём лингвистические переменные в подзаключениях попарно не равны друг другу, то степень истинности каждого из подзаключений равна алгебраическому произведению соответствующего значения  $b_i''$  на весовой коэффициент  $F_i$ .

В результате находим все значения  $c_k$  степеней истинности подзаключений для каждого из правил  $R_k$ , входящих в рассматриваемую базу правил  $P$  системы нечёткого вывода. Весовой коэффициент  $F_i$  задают таким образом, чтобы он приводил величину  $c_k$ , в зависимости от того, превышает её значение 0,5 или нет, к двум значениям: 1 (да, превышает) или 0 (нет, не превышает). Это множество значений обозначим через  $\mathbf{C}=\{c_1, c_2, \dots, c_q\}$ , где  $q$  – общее количество подзаключений в базе правил.

После нахождения множества  $\mathbf{C}=\{c_1, c_2, \dots, c_q\}$  определяются функции принадлежности каждого из подзаключений для рассматриваемых выходных лингвистических переменных. Для этой цели использован метод:

$$\circ \text{ average-активизация: } \mu'(y) = 0,5 \cdot (c_i + \mu(y)), \quad (3.2.3)$$

где  $\mu(y)$  – функция принадлежности терма, который является значением некоторой выходной переменной  $\omega_j$ , заданной на универсуме  $Y$ . При обучении данная функция известна и задаётся заранее.

Этап активизации считается законченным, когда для каждой из выходных лингвистических переменных, входящих в отдельные подзаключения правил нечётких продукций, будут определены функции принадлежности нечётких множеств их значений, т.е. совокупность нечётких множеств:  $C_1, C_2, \dots, C_q$ , где  $q$  – общее количество подзаключений в базе правил системы нечёткого вывода. Таким образом, на этом этапе происходит выбор правил системы нечёткого вывода для каждого заключения.

4. Цель *аккумуляции* заключается в том, чтобы объединить или аккумулировать все степени истинности заключений (подзаключений) для получения функции принадлежности каждой из выходных переменных множества  $W = \{\omega_1, \omega_2, \dots, \omega_s\}$ . Причина необходимости выполнения этого этапа состоит в том, что подзаключения, относящиеся к одной и той же выходной лингвистической переменной, принадлежат различным правилам системы нечёткого вывода. До начала этого этапа предполагаются известными значения истинности всех подзаключений для каждого из правил  $R_k$ , входящих в рассматриваемую базу правил  $P$  системы нечёткого вывода, в форме совокупности нечётких множеств:  $C_1, C_2, \dots, C_q$ , где  $q$  – общее количество подзаключений в базе правил системы нечёткого вывода. Далее последовательно рассматривается каждая из выходных лингвистических переменных  $\omega_j \in W$  и относящиеся к ней нечёткие множества:  $C_{j1}, C_{j2}, \dots, C_{jq}$ . Результат аккумуляции для выходной лингвистической переменной  $\omega_j$  определяется как объединение нечётких множеств  $C_{j1}, C_{j2}, \dots, C_{jq}$ .

Этап аккумуляции считается законченным, когда для каждой из выходных лингвистических переменных будут определены итоговые функции принадлежности нечётких множеств их значений, т.е. определена структура нечёткого дерева решений и совокупность нечётких множеств:  $C_1', C_2', \dots, C_s'$ , где  $s$  – общее количество выходных лингвистических переменных в базе правил системы нечёткого вывода.

5. Цель *дефаззификации* заключается в том, чтобы, используя результаты аккумуляции всех выходных лингвистических переменных, получить обычное количественное значение каждой из выходных переменных, которое может быть использовано специальными устройствами, внешними по отношению к системе нечёткого вывода. До начала этого этапа сформированы функции принадлежности всех выходных лингвистических переменных в форме нечётких множеств:  $C_1', C_2', \dots, C_s'$ , где  $s$  – общее количество выходных лингвистических переменных в базе правил системы нечёткого вывода. Далее последовательно рассматривается каждая из выходных лингвистических переменных  $\omega_j \in W$  и относящееся к ней нечёткое множество  $C_j'$ . Результат дефаззификации выходной лингвистической переменной  $\omega_j$  определяется в виде количественного значения  $y_j \in \mathbb{R}$ . На этом этапе получаем оценку качества для выполнения распознавания параметров движущихся объектов обучающей выборки определением значения целевой функции (ошибка распознавания объектов по параметрам).
6. *Процесс построения нечёткого дерева решений* заключается в оптимизации функции принадлежности всех входных и выходных лингвистических переменных при повторении выше описанных блоков до тех пор, пока не будет достигнуто определённое значение целевой функции, то есть заданное минимальное количество ошибочно найденных значений искомых параметров для обучающей выборки.

### 3.2.2 Результаты обработки данных производственной нечёткой системой в виде нечёткого дерева решений

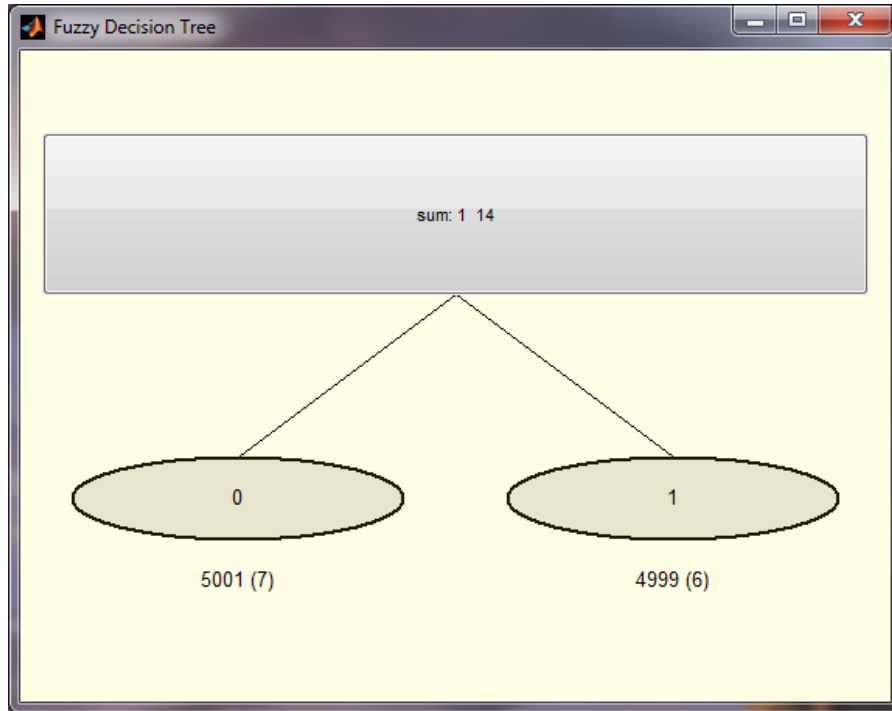
Результаты обработки данных производственной нечёткой системой в виде нечёткого дерева решений представлены в таблице 3.7. [177, 178].

Таблица 3.7. Результаты обработки реальных данных нечётким деревом решений

Параметр движущегося объекта	Ошибка по обучающей выборке (кол-во из 10000 примеров), %		Ошибка по тестовой выборке (кол-во из 200 примеров), %	
	По признакам	По главным компонентам	По признакам	По главным компонентам
Класс	0,13 (13)	0,47 (47)	0	0,5(1)
Направление движения	0,25 (25)	0,51 (51)	0,5 (1)	0
Масса, кг	4,75 (475)	6,65 (665)	2,5 (5)	10 (20)
Длина пути, м	8,95 (895)	14,89 (1489)	13 (26)	25,5 (50)

Нечёткие деревья решений для «класса» по признакам и главным компонентам показаны на рисунках 3.22(a) и 3.23(a). Аналогичную структуру имеет нечёткое дерево решений по признакам для «направления движения», как видно на рисунке 3.22(б). В каждом из них содержится один узел и две ветви решения. Структура дерева по главным компонентам при определении направления движения оказалась несколько сложнее (рис. 3.23(б)). В системе распознавали два вида объектов (катящийся и шагающий) и два направления движения по ВОИС. Полученные виды функций принадлежности для признаков и главных компонент, используемых для распознавания класса и направления движения, представлены на графиках (рис. 3.24 (a, б и в), 3.25 (a, б), 3.26 (a, б, в и г)). Для массы и длины пути нечёткие деревья решений имеют более сложную ветвистую структуру и содержат 17 и 36 узлов по признакам, а по главным компонентам 28 и 63 узлов, соответственно.

а)



б)

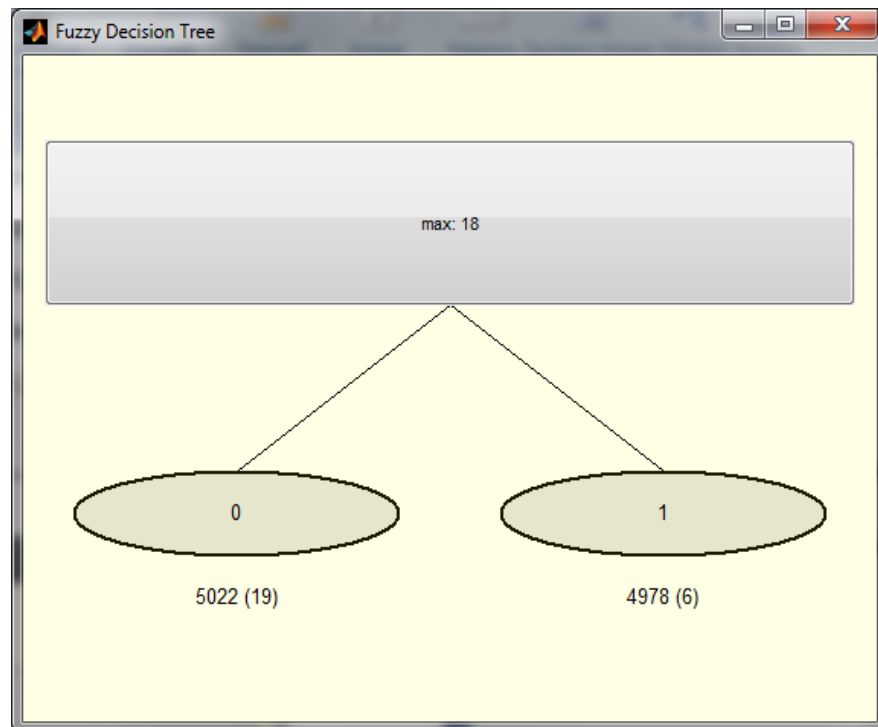
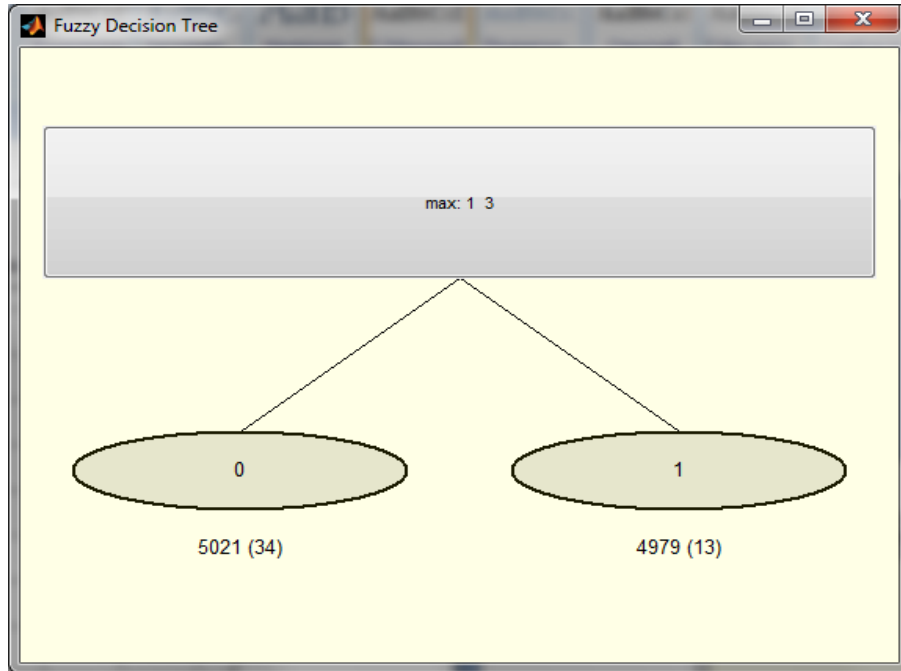


Рис. 3.22. Структура нечёткого дерева решений по признакам для: а) «класса»; б) «направления движения»

а)



б)

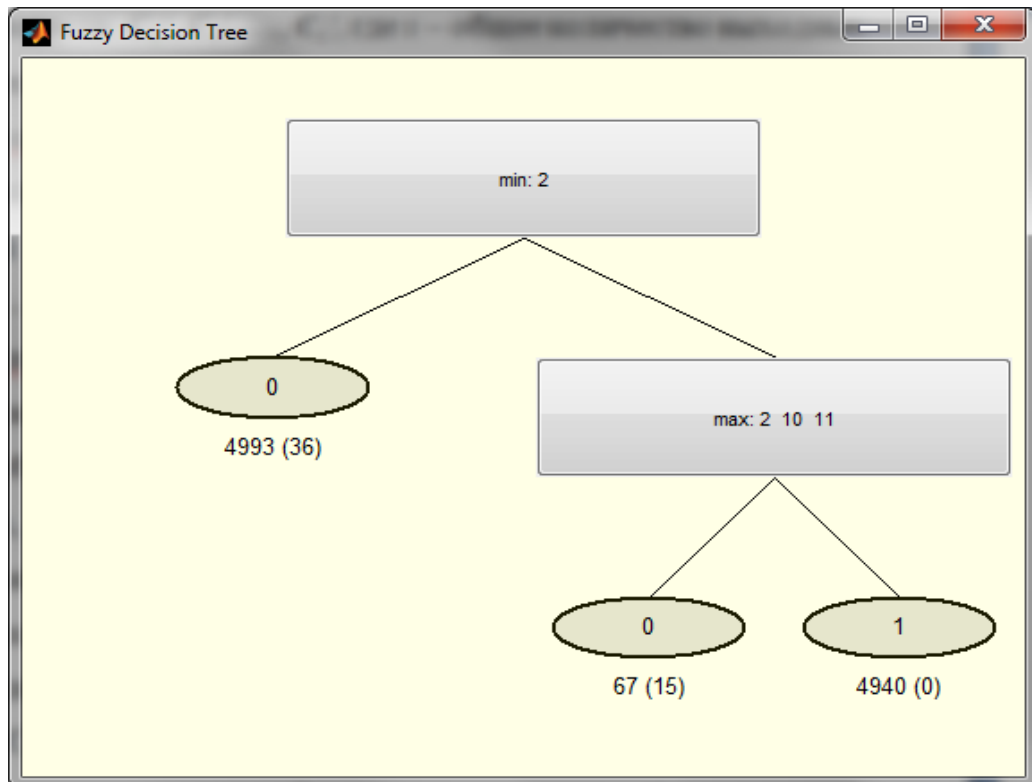
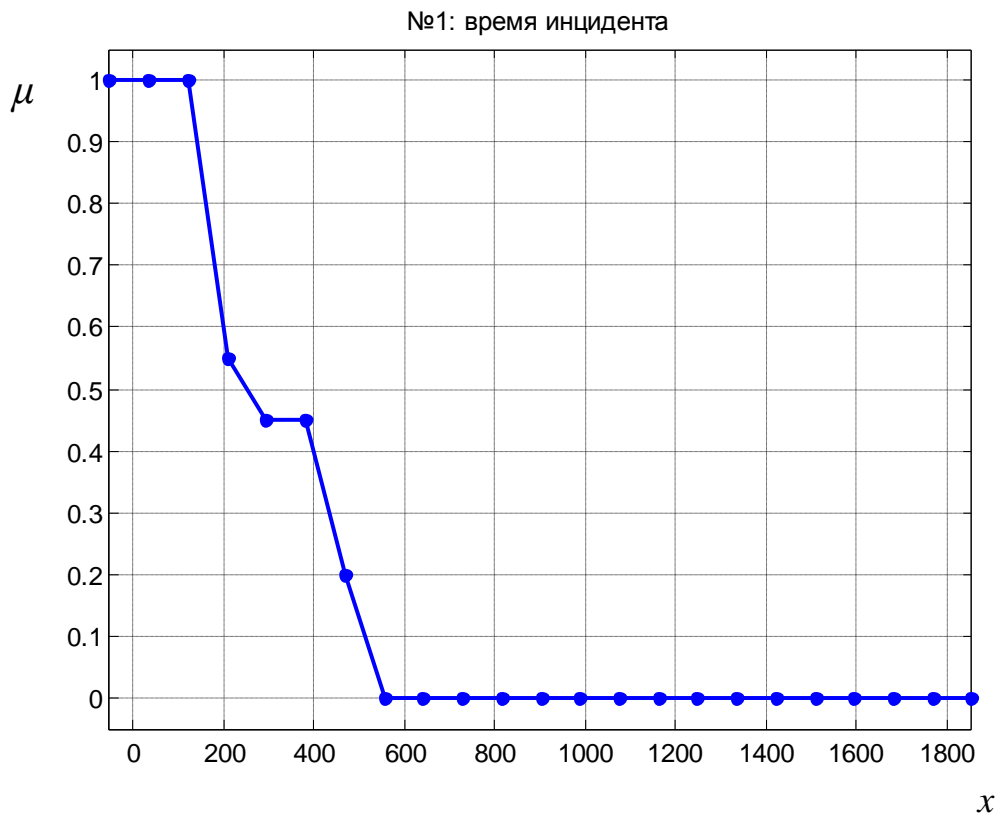
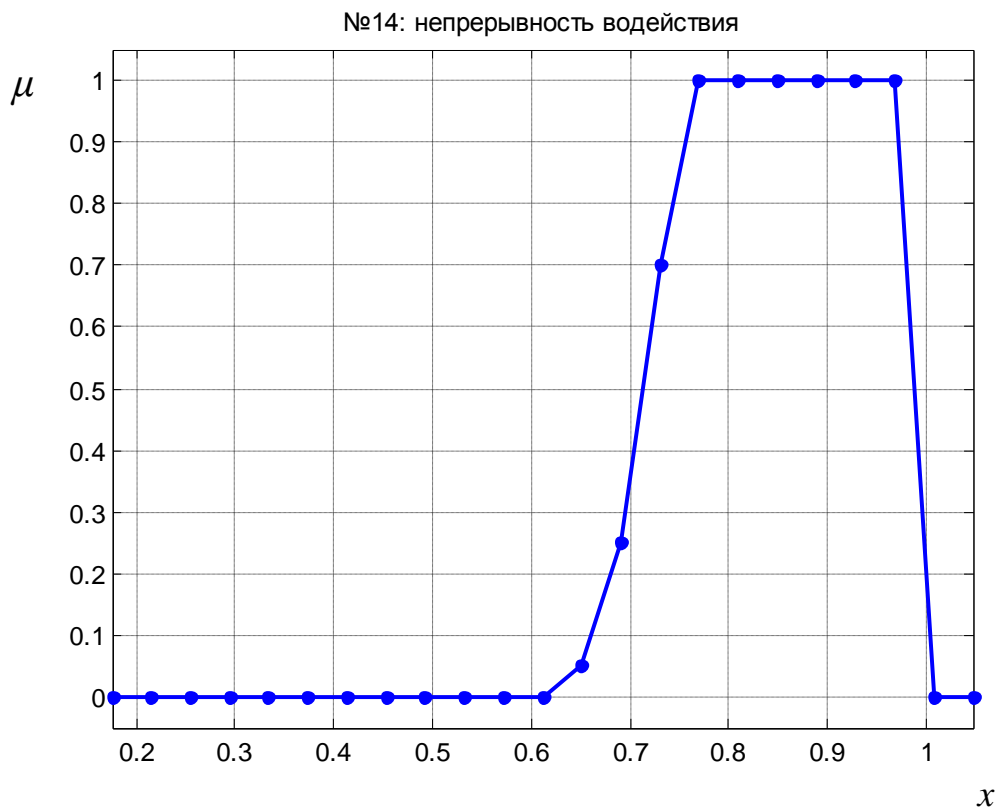


Рис. 3.23. Структура нечёткого дерева решений по главным компонентам для: а) «класса»; б) «направления движения»

а)



б)



в)

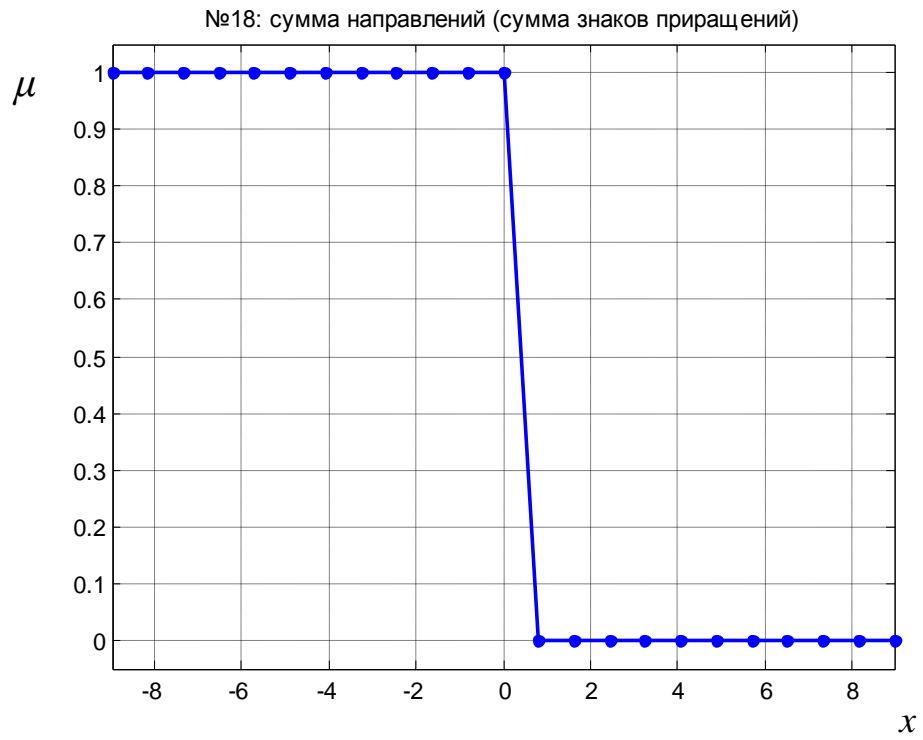
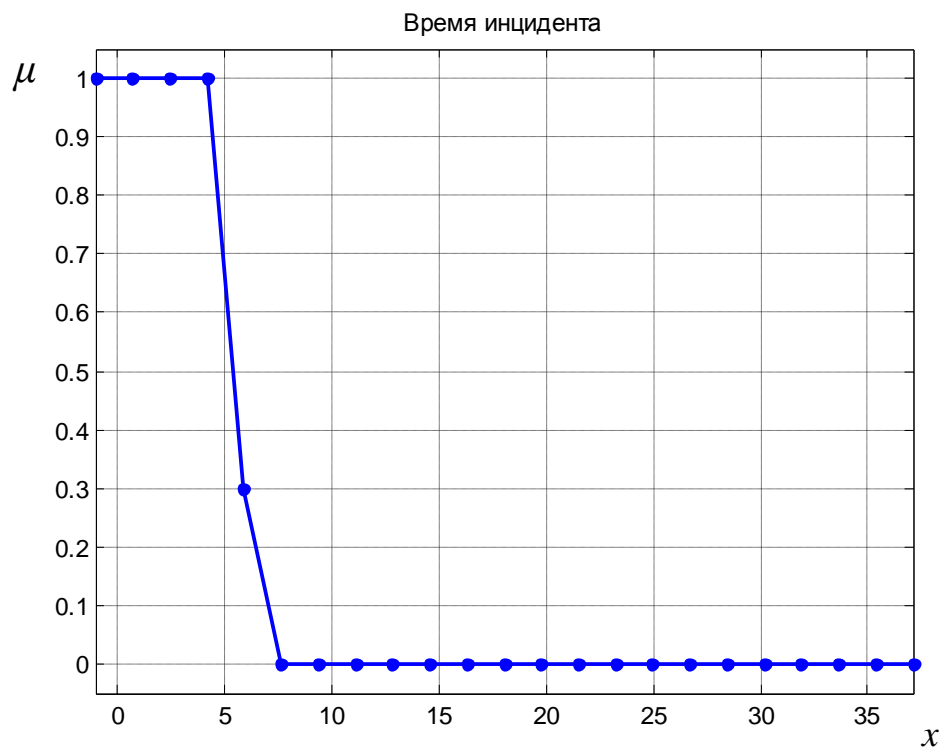


Рис. 3.24. Функции принадлежности: а) для признака № 1 при определении класса; б) для признака № 14 при определении класса; в) для признака №18 при определении направления движения

а)





б)

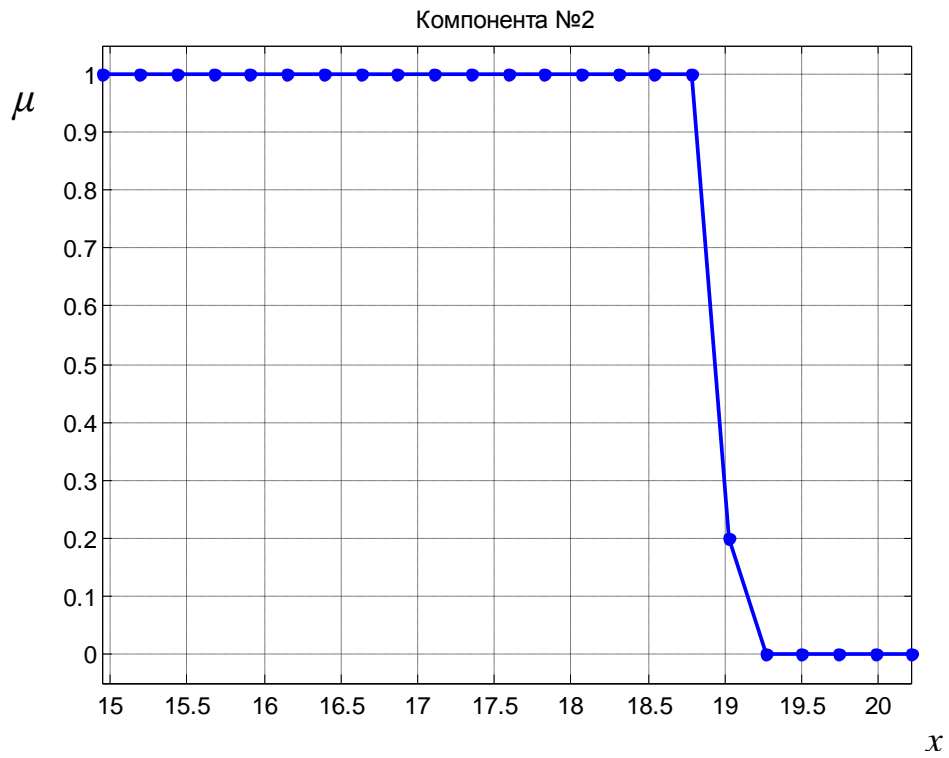
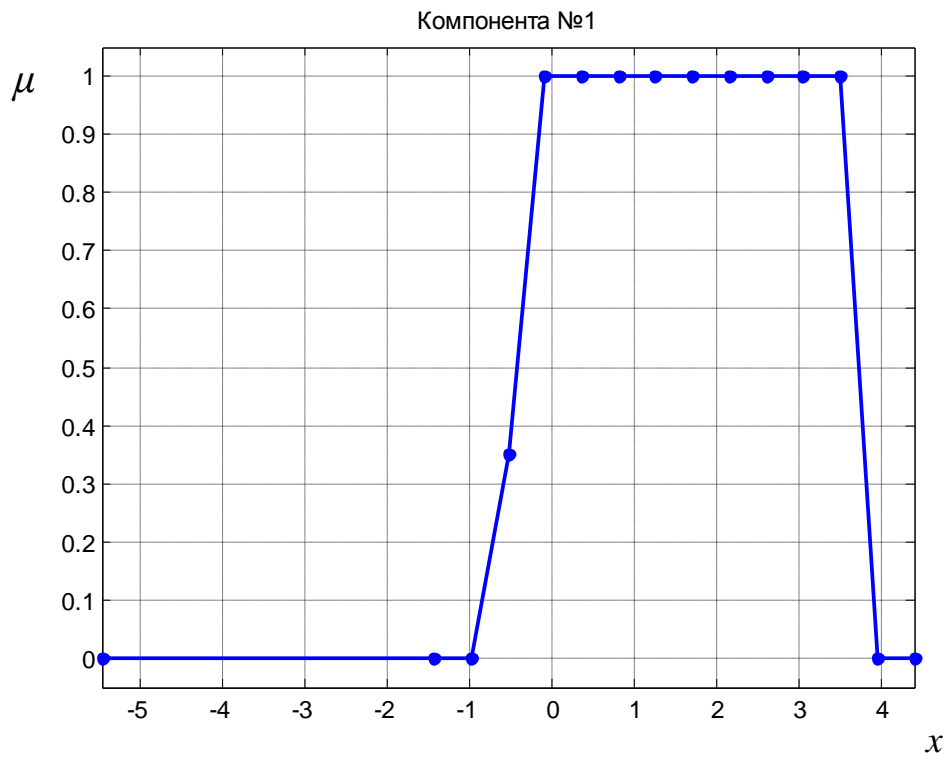
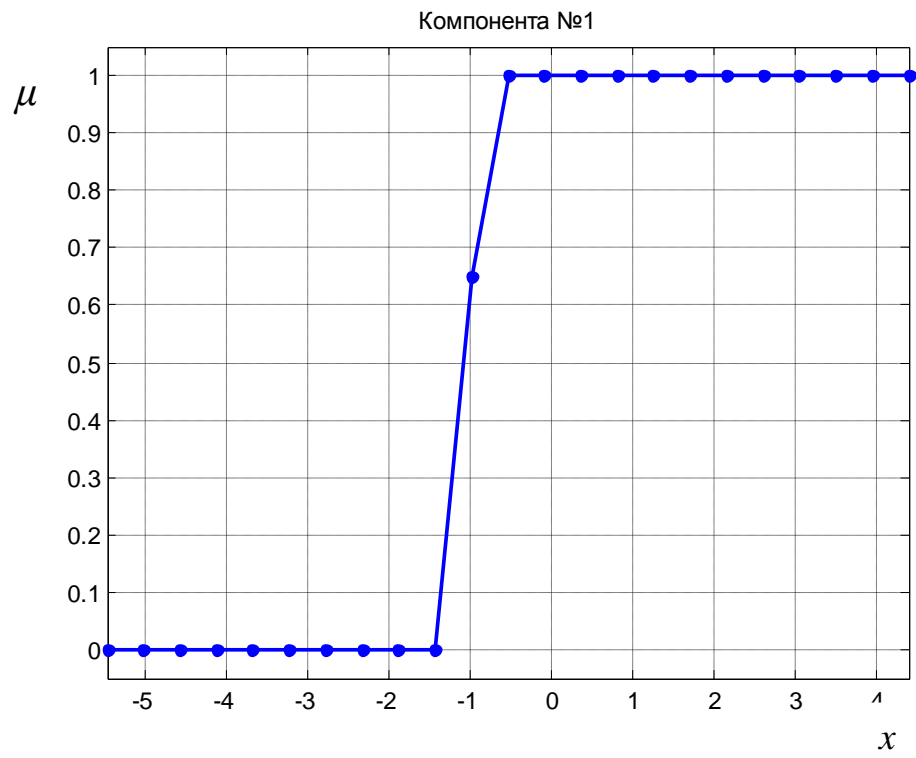


Рис. 3.25. Функции принадлежности при определении класса: а) для времени инцидента; б) для главной компоненты №2

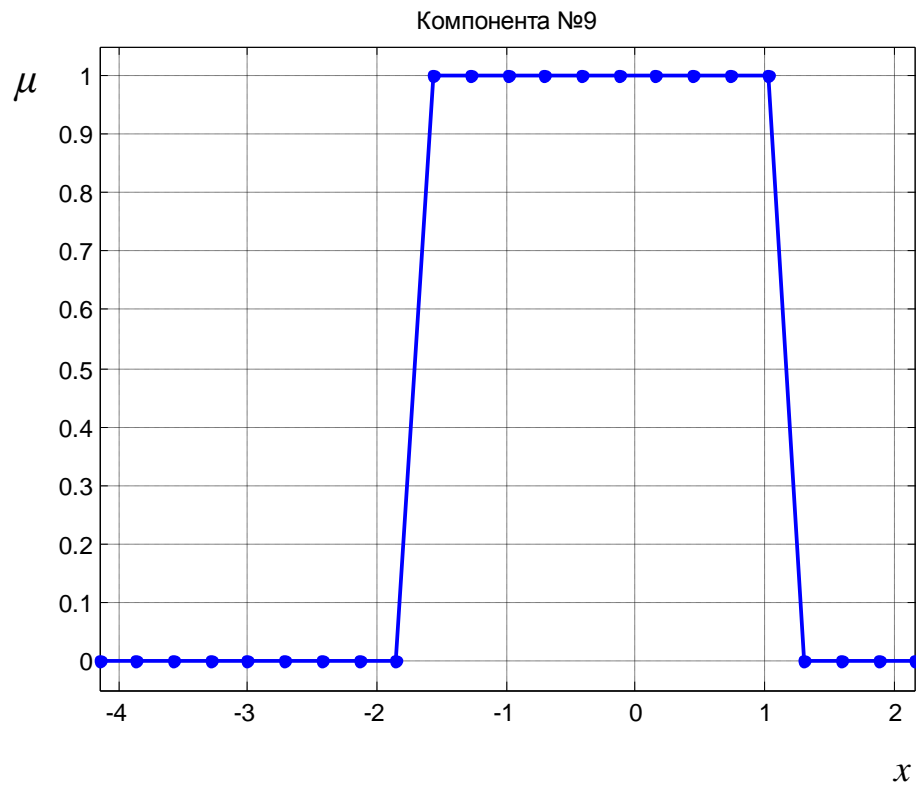
а)



б)



в)



з)

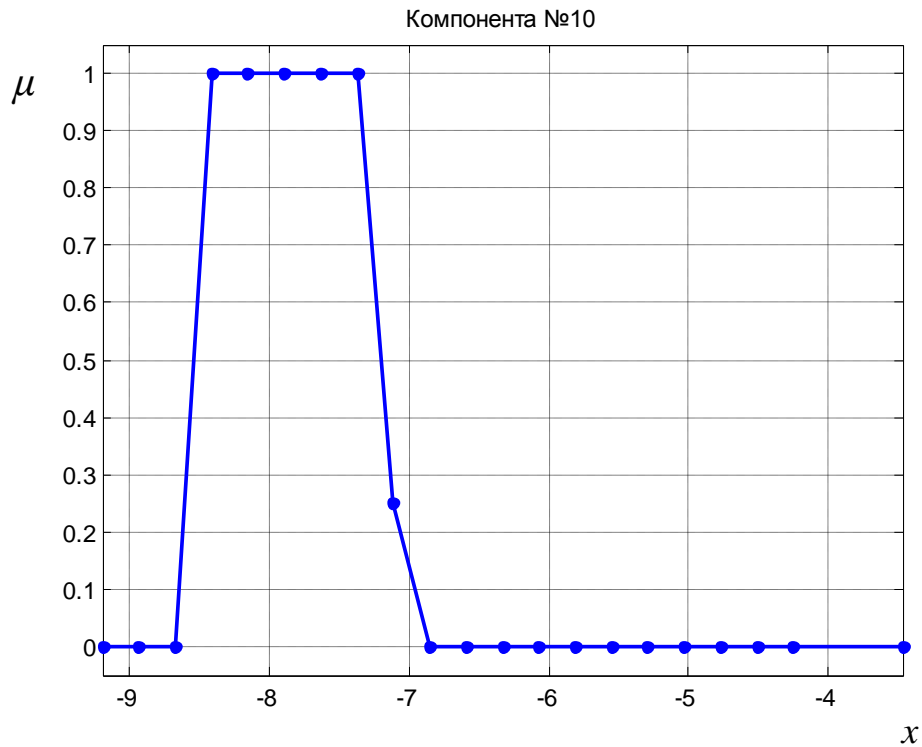


Рис. 3.26. Функции принадлежности при определении направления движения: а) для главной компоненты №1 первого узла; б) для главной компоненты №1 второго узла в) для главной компоненты № 9 второго узла; з) для главной компоненты № 10 второго узла

Критерий качества работы разработанной продукционной нечёткой системы в виде нечёткого дерева решений (таблица 3.7.) показывает, что такая система способна решать поставленную задачу с удовлетворительной точностью. Наибольшее количество ошибок было допущено при определении массы и длины пути для тестовой выборки они составили 6,65% и 14,89% по признакам, а по главным компонентам 10% и 25,5%, соответственно. Причём, в случае использования главных компонент, ошибки распознавания для этих параметров оказались больше. Надо отметить, что структура получаемого нечёткого дерева решений будет усложняться по мере возрастания количества распознаваемых объектов и зависит от типа определяемых параметров, поэтому такая система для обработки данных может быть использована при распознавании ограниченного числа классов объектов и их определяемых параметров.

### 3.3 ВЫВОДЫ

Основным итогом выполненной работы стала реализация и обоснование комплексного подхода к созданию ИРИИС на основе РВОИС сегментарного типа, предназначенного для распознавания динамических образов и параметров движущихся объектов, с возможностью расширения предметной области решаемых задач по распознаванию динамических образов. Для выполнения главной задачи исследований в работе было проведено моделирование динамических образов и создан программно-алгоритмический комплекс, позволяющий исследовать, тестировать предлагаемые подходы, методы и способы обработки данных и доказывать их применимость на практике. Предложена методологическая база построения подобных интеллектуальных информационно-измерительных систем.

Разработанные модули для генерации выборки сенсорных данных и для преобразования измеренных данных с макета ВОИС охраны периметра дают возможность подготовить выборки примеров данных для обработки, по результатам которой проверяют предложенные подходы и оценивают эффективность работы системы. Один подход основан на обработке данных многослойными нейронными сетями прямого распространения, а другой на проектировании продукционной нечёткой системы в виде нечёткого дерева решений. Проблема снижения объема (сжатия) массива входных данных была решена тремя способами: с помощью сохранения фиксированного объема информации при обработке и её обновлением в режиме реального времени, определением главных компонент и выделением значимых признаков. Процедуры по нахождению главных компонент и признаков привели к формированию двух моделей образов, которые использовались для данных, полученных с генератора выборки сенсорных данных и с макета ВОИС охраны периметра после обработки с помощью модуля преобразования измеренных данных.

Таким образом, предобработка данных и их преобразование предлагаемыми способами даёт возможность значительно ускорить процесс обработки, как

нейронными сетями, так и нечётким деревом решений и позволяет решать задачу идентификации движущихся объектов в зоне мониторинга ВОИС практически без потерь в точности. Тестирование СОД ИРИИС и предложенные оценки эффективности работы ИРИИС по критериям качества работы системы показывают, что способ предобработки и модель образа по данным должны выбираться в системе в зависимости от качества обучения разными подходами и сложности формирования моделей распознаваемых динамических образов и с учётом факторов влияния внешней среды.

## **ЗАКЛЮЧЕНИЕ**

В диссертации изложены подходы, способы и методы создания ИРИИС на основе РВОИС сегментарного типа для распознавания динамических образов движущихся объектов в зоне мониторинга сети.

На основе изложенных материалов можно сделать следующие основные выводы:

1. Разработаны обобщённая, функциональная и структурная модели ИРИИС на основе РВОИС сегментарного типа для распознавания динамических образов в зоне мониторинга измерительной сети, составляющие базу для проектирования в дальнейшем системы поддержки принятия решений.
2. Предложен новый подход при построении ИРИИС на основе РВОИС сегментарного типа с учетом специфики ЭС как отдельного класса интеллектуальных информационно – измерительных систем.
3. Разработан набор специальных способов и методов обработки данных в ИРИИС на основе РВОИС сегментарного типа для идентификации движущихся объектов в зоне мониторинга измерительной сети.
4. Предложен метод проектирования продукционной нечёткой системы в виде нечёткого дерева решений, не требующий от эксперта специальной подготовки и позволяющий достаточно легко строить модель базы знаний через обучение.

5. Создан программно-алгоритмический комплекс и проведено моделирование динамических образов для реализации подходов, способов и методов обработки данных в виде продукционной нечёткой системы и экспертной системы с привлечением нейронных сетей, что представляет собой основу предложенной ИРИИС и позволяет описать её функционирование и структуру.
6. Предложена методологическая база построения подобных интеллектуальных информационно-измерительных систем, включающая подходы к созданию ИРИИС на основе РВОИС, модули для генерации выборки сенсорных данных и для преобразования измеренных данных с макета ВОИС охраны периметра, способы и методы для формирования моделей образов по выделенным признакам и главным компонентам и экспериментальный макет ВОИС охраны периметра, предоставляющий основу для разработки ЭС мониторинга в области систем безопасности.

**СПИСОК ЛИТЕРАТУРЫ**

1. Кульчин, Ю.Н. Распределенные волоконно-оптические измерительные системы / Ю.Н. Кульчин – М.: Физматлит, 2001. – 272 с.
2. Кульчин, Ю.Н. Распределённые адаптивные оптоэлектронные информационно-измерительные системы / Ю.Н. Кульчин, О.Б. Витрик, К.П. Урываев // Вестник ДВО РАН, 2005. - №6. – С. 66-76.
3. Кульчин, Ю.Н. Волоконно-оптическая измерительная сеть для регистрации параметров колебательных процессов / Ю.Н. Кульчин, О.Б. Витрик, К.П. Горбачев, А.П. Аносов, О.В. Кириченко, Ю.С. Петров, О.Т. Каменев // Измерительная техника, 1995. – Т. 3. – С. 32-35.
4. Кульчин, Ю.Н. Восстановление векторных физических полей с использованием двумерной волоконно-оптической измерительной сети / Ю.Н. Кульчин, О.Б. Витрик, О.Т. Каменев, Р.В. Ромашко // Измерительная техника, 1999. – Т.6. – С. 21-28.
5. Кульчин, Ю.Н. Восстановление физических полей с использованием двумерной волоконно-оптической измерительной сети / Ю.Н. Кульчин, О.Б. Витрик, О.В. Кириченко // Измерительная техника, 1999. – Т. 3. – С. 24-30.
6. Кульчин, Ю.Н. Метод обработки сигналов двухмодового волоконного интерферометра / Ю.Н. Кульчин, О.Б. Витрик, О.В. Кириченко, Ю.С. Петров, О.Т. Каменев // Автометрия, 1995. – №5. – С. 32-35.
7. Kulchin, Yu.N. Principal of reconstruction of the single influences on fiber-optical measuring network / Yu.N. Kulchin, E.V. Denisova, I.V. Denisov, N.A.Rybalchenko // Pacific Science Review, 2003. – Vol. 5 (1). – P. 32-37.
8. Кульчин, Ю.Н. Принцип организации волоконно-оптической измерительной сети повышенной размерности / Ю.Н. Кульчин, И.В. Денисов, В.А. Седов, О.В. Кириченко, Е.В. Денисова (Закасовская), Р.С. Дроздов // Оптико-электронные информационно-энергетические технологии, 2002. – Т. № 2 (4). – С. 205-211.

9. Denisov, I.V. Model of optoelectronic measuring intelligent system / I.V. Denisov, Y.N. Kulchin, O.V. Kirichenko, V.A. Sedov, R.S. Drozdov, E.V. Denisova // Proc. of International Conference Physics and Control, 2003. – Т. 1. – С. 172-175.
10. Кульчин, Ю.Н. Принципы организации матриц связей оптической нейронной сети на объемных оптических элементах / Ю.Н. Кульчин, И.В. Денисов, Е.В. Денисова (Закасовская) // Нейрокомпьютеры: разработка, применение, 2003. – № 7. – С. 21-27.
11. Denisov, I.V. Principal of reconstruction of the ternary influences on the fiber-optical measuring network / I.V. Denisov, E.V. Denisova, N.A. Rybalchenko // Proc. of Fourth Asia-Pacific Conference “Fundamental Problems of Optoelectronics and Microelectronics”, 2004. – Т. 1. – С. 238-240.
12. Denisov, I.V. Reconstruction of twice influences on the fiber-optical measuring network / I.V. Denisov, E.V. Denisova, N.A. Rybalchenko // Труды II Международной конференции «Параллельные вычисления и задачи управления», 2004. – С. 531-534.
13. Гридин, А.А. Обработка информации с распределенного температурного поля / А.А. Гридин, Е.В. Денисова, И.В. Денисов // Proc. of 7-th International Conference Digital Signal Processing and its Applications, 2005. – С. 365-368.
14. Левин, Г.Г., Вишняков, Г.Н. Оптическая томография. / Г.Г. Левин, Г.Н. Вишняков – М.: «Радио и связь», 1989. – 224 с.
15. Котов, О.И. О применении томографических методов в волоконно-оптических датчиках / О.И. Котов // Письма в ЖТФ, 1990. – Т. 16, 2. – С. 90-94.
16. Ауслендер, А.Л. Решение интегрального уравнения Радона в оптическом процессоре / А.Л. Ауслендер, Г.Н. Вишняков, Г.Г. Левин // Оптика и спектроскопия, 1980. – Т. 49, № 5. – С. 946-951.
17. Ауслендер, А.Л. О восстановлении изображений поперечных сечений объектов по их проекциям / А.Л. Ауслендер, Г.Н. Вишняков, Г.Г. Левин // Оптика и спектроскопия, 1981. – Т. 50, №2. – С. 408-411.



18. Кульчин, Ю.Н. Восстановление векторных физических полей методом оптической томографии / Ю.Н. Кульчин, О.Б. Витрик, О.В. Кириченко, Ю.С. Петров, О.Т. Каменев // Квантовая электроника, 1995. – Т. 22, 10. – С. 1009-1012.
19. Kamenev, O.T. Training two-layer neural network model for tomography data processing / O.T. Kamenev // Proc. of Int. Conf. OCEAN'95. – San-Diego, 1995. – С. 2086-2087.
20. Кульчин, Ю.Н. Томографические методы для исследования векторных полей при помощи волоконно-измерительных систем / Ю.Н. Кульчин, О.Б. Витрик, Р.В. Ромашко, Ю.С. Петров, О.В. Кириченко // Квантовая электроника, 1997. – Т.24, №5. – С. 467-470.
21. Kulchin, Yu.N. Tomography methods for vector fields study by using space distributed fiber optic sensors with integral sensitivity / Yu.N. Kulchin, O.B. Vitrik, R.V. Romashko, Yu.S. Petrov, O.V. Kirichenko, O.T. Kamenev // Fiber and Integrated Optics, 1998. – Vol. 17.
22. Denisov, I.V. Definition of the fiber-optical tomography problem / I.V. Denisov, E.V. Denisova, N.A. Rybalchenko, V.A.Sedov // Proc. of International Conference Physics and Contro, 2005. – Т. 1. – С. 826-829.
23. Kulchin, Yu.N. Distributed fiber-optic acoustic sensor / Yu.N. Kulchin, O.B. Vitrik, O.V. Kirichenko, Yu.S. Petrov, O.T. Kamenev // Proc. of Distributed and multiplexed fiber optic sensor IV Conf., 1994. – Vol. 2294. – P. 129-132
24. Merzbacher, C. Fiber-optic sensors in concrete structures: a review / C. Merzbacher, A. Kersey, E. Friebele // Smart Mater. Struct, 1996. – Vol. 5, №2. - P. 196-208.
25. Kulchin, Yu.N. Distribute fiber-optic sensor for seismoacoustic investigation / Yu.N. Kulchin, O.B. Vitrik, O.V. Kirichenko, Yu.S. Petrov // Proc. 3-rd International Russian Fiber Optic Conference, 1993. – Vol. 2. – P. 291-294.
26. Kulchin, Yu.N. Concept of the neural network model realizing for tomography data processing / Yu.N. Kulchin, O.B. Vitrik, O.V. Kirichenko, Yu.S. Petrov, O.T. Kamenev, O.G. Maksaev // Proceedings of International Conf. on Holography and Correlation Optics, 1995. – Т. 2647. – P. 150-152.

27. Kulchin, Yu.N. Holographic neural network for processing of signals of distributed optical fiber measuring networks with the tomography principle of data gathering / Yu. Kulchin, O. Vitrik, Yu. Petrov, O. Kirichenko, O. Kamenev, R. Romashko, I. Denisov // *Optical Memory and Neural Networks*. 1997. – Vol. 6, 2. –P. 149-156.
28. Кульчин, Ю.Н. Оптоэлектронная нейроподобная система обработки выходных данных волоконно-оптической измерительной сети / Ю.Н. Кульчин, И.В. Денисов, О.Т. Каменев // *Письма ЖТФ*, 1999. – Т.25, вып. 6. – С. 65-69.
29. Kulchin, Yu.N. Computer neural networks for processing of optical tomography information / Yu. Kulchin, I. Denisov, V. Obuh, O. Kamenev, O. Vitrik, Yu. Petrov, R. Romashko // *Pacific Science Review*, 1999. – Vol. 1. – P. 1-4.
30. Denisov, I.V. Principales of organization of neural-like system on the basis of a matrix of photoelectric cells *Optoelectronic Information* / I. Denisov, Yu. Kulchin, V. Obuh, O. Kamenev // *Systems and Processing*, 2001. – Vol. 4513. – P. 52-57.
31. Kulchin, Yu.N. Neural processing system for optical information measuring systems distributed fiber optical sensors and measuring networks / Yu.N. Kulchin, O.T. Kamenev, I.V. Denisov // *Proc. of SPIE*, 2001. – Vol. 4357. – P. 109-117.
32. Kulchin, Yu.N. Fiber-Optic Distributed Measuring System with Neural Network Technology of Signal Array Processing / Yu.N. Kulchin, I.V. Denisov, E.V. Denisova, V.A. Sedov, R.S. Drozdov // *International workshop on Bio-Signal Analysis and its Applications*, 2002. – Vol. 2. – P. 3-6.
33. Денисов И.В. Применение приближенных алгебраических и нейросетевых методов решения томографической задачи / И.В. Денисов, Е.В. Денисова // *Электронный журнал «Исследовано в России»*, 2002. – Т. 201. – С. 2222–2228.
34. Кульчин, Ю.Н. Многомерная обработка сигналов с использованием волоконно-оптической измерительной сети / Ю.Н. Кульчин, О.Б. Витрик, О.В. Кириченко, Ю.С. Петров // *Квантовая электроника*, 1993. – Т. 20, №5. – С. 711-714.

35. Kulchin, Yu.N. Principal of reconstruction of the single influences on fiber-optical measuring network / Yu.N. Kulchin, E.V. Denisova, I.V. Denisov and N.A. Rybalchenko // Pacific Science Review, 2003. – T.1, № 5. – P. 32-37.
36. Cichocki, A. Neural networks for linear inverse problems with incomplete data especially in applications to signal and image reconstruction / A. Cichocki, R. Unbehauen, M. Lendl, K. Weinzierl // Neurocomputing, 1995. – Vol.8. – P. 7-41.
37. Denisov, I.V. Neural data processing method for fiber-optic distributed measuring systems / I.V. Denisov, O.T. Kamenev, A.Yu. Kim, Yu.N. Kulchin, A.V. Panov // Optical Memory & Neural Networks (Information Optics), 2003. – T12, №3. – P. 165-172.
38. Kulchin, Yu.N. Application of algebraic and neural-like methods for reconstruction of distribution functions of physical fields / Yu.N. Kulchin, E.V. Denisova, I.V. Denisov // Optical Memory & Neural Networks (Information Optics), 2003. – Vol. 12, №4. – P. 283-297.
39. Кульчин, Ю.Н. Обучающаяся нейросеть для обработки томографических данных / Ю.Н. Кульчин, О.Т. Каменев // Кибернетика и вуз. Томск: ТПУ, 1994. – №28. – С. 3-7.
40. Kulchin, Yu.N. Neural network for reconstruction of signal from distributed measuring system of optical amplitude sensors / Yu.N. Kulchin, A.V. Panov // Pacific Science Review, 2001. – Vol. 3. – P. 1-4.
41. Muller, B. Neural Networks / B. Muller, J. Reinhardt - Berlin: Springer-Veterad, Heindelberg. 1990.
42. Розенблат, Ф. Принципы нейродинамики. Перцептрон и теория механизмов мозга / Ф. Розенблатт – М.: «Мир», 1965. – 480 с.
43. Rosenblatt, F. The Perceptron: A Probabilistic model for information storage and organization in the brain / F. Rosenblatt // Cornell Aeronautical Laboratory, Psychological Review, 1958. – Vol. 65, 6. – P. 386–408.
44. Галушкин, А.И. Теория нейронных сетей / А.И. Галушкин - М.: Издательское предприятие редакции журнала «Радиотехника», 2000. – 485 с.

45. Саймон, Хайкин. Нейронные сети: полный курс. (Neural Networks) – 2-е изд., исправленное / Хайкин Саймон – М.: «Вильямс», 2006. – 1103 с.
46. Комарцова, Л.Г. Нейрокомпьютеры: учебное пособие для вузов / Л.Г. Комарцова, А.В. Максимов – М.: Изд-во МГТУ им. Н.Э. Баумана, 2002. – 320 с.
47. Калан, Р. Основные концепции нейронных сетей / Р. Калан – М.: Изд. дом «Вильямс», 2001. – 288 с.
48. Оссовский, С. Нейронные сети для обработки информации / С.Оссовский – М.: «Финансы и статистика», 2002. – 344 с.
49. Kulchin, Yu.N. Dynamic Image Pattern Recognition Using the Reconstructed Signal of Fiber-Optic Measuring System / Yu.N. Kulchin, A.Yu. Kim // Pacific Science Review, 2002. – Vol.4. – P. 55-58.
50. Кульчин Ю.Н., Ким А.Ю. Распознавание динамического объекта по восстановленному сигналу волоконно-оптической измерительной системы / Ю.Н. Кульчин, А.Ю. Ким // Материалы НТК «Молодёжь и научно-технический прогресс», ДВГТУ, г. Владивосток, 2004. – С. 111.
51. Кульчин Ю.Н., Ким А.Ю. Распознавание динамических образов распределённой информационно-измерительной системой сегментарного типа / Ю.Н. Кульчин, А.Ю. Ким // Проблемы управления, Москва, 2006. - №5. – С. 52-57.
52. Каляев, И.А. Однородные нейроподобные структуры в системах выбора действий интеллектуальных роботов / И.А. Каляев и А.Р. Гайдук – М.: «Янус-К», 2000. – 346 с.
53. Городецкий, А.Е. Программные средства интеллектуальных систем / А.Е. Городецкий, В.В. Дубаренко, И.Л. Тарасова, А.В. Шереверов – СПб.: Изд-во СПбГТУ, 2000. – 171с.
54. Денисов, И.В. Архитектуры систем искусственного интеллекта. / И.В. Денисов – Вл-к.: Изд-во МГУ им. Г.И. Невельского, 2004. – 231с.
55. Ким, Дж.-О. Факторный анализ: статистические методы и практические вопросы / сборник работ «Факторный, дискриминантный и кластерный анализ»:

- пер. с англ. Под редакцией И.С. Енюкова. / Дж.-О. Ким, Ч. У. Мьюллер — М.: «Финансы и статистика», 1989. – 215 с.
56. Афифи, А. Статистический анализ: Подход с использованием ЭВМ / А. Афифи, С. Эйзен — М.: «Мир», 1982. – 488 с.
57. Дрейпе, Н. Прикладной регрессионный анализ. Множественная регрессия. (Applied Regression Analysis) – 3-е изд. / Н. Дрейпер, Г. Смит – М.: «Диалектика», 2007. – 912 с.
58. Фёрстер, Э. Методы корреляционного и регрессионного анализа. (Methoden der Korrelation-und Regressiolyns analyse). / Э. Фёрстер, Б. Рёнц – М.: «Финансы и статистика», 1981. – 302 с.
59. Стренг, Г. Линейная алгебра и ее применения / Г. Стренг – М.: «Мир», 1980. – 459 с.
60. Каханер, Д. Численные методы и программное обеспечение / Д. Каханер, К. Моулер, С. Нэш – М.: «Мир», 1998. – 575 с.
61. Уинстон, П. Искусственный интеллект / П. Уинстон - М.: «Мир», 1980. – 519 с.
62. Кузнецов, В.Е. Представление в ЭВМ неформальных процедур: продукционные системы / В.Е. Кузнецов [С послесловием Д.А. Поспелова] – М.: Наука. Гл. ред. физ.-мат. лит. – 1989. – (Проблемы искусственного интеллекта) – 160 с.
63. Джексон, П. Введение в экспертные системы (Introduction to Expert Systems) – 3-е изд. / Питер Джексон — М.: «Вильямс», 2001. – 624 с.
64. Уотермен, Д. Руководство по экспертным системам: Пер. с англ. под ред. В.Л. Стефанюка / Д. Уотермен — М.: «Мир», 1989. – 388 с.
65. Гаврилова Т.А. Базы знаний интеллектуальных систем. Учебник / Т.А. Гаврилова, В.Ф. Хорошевский – СПб.: «Питер», 2000. – 384 с.
66. Бураков, М.В. Нейронечёткие системы управления / М.В. Бураков, А.С. Коновалов // Информационно-управляющие системы, 2002. - №1. – С. 2-7.

67. Takagi, T., Sugeno, M. Fuzzy identification of systems and its application to modeling and control / T. Takagi, M. Sugeno // IEEE Trans.-SMS, 1985. – P. 116-132.
68. Рыжов, А.П. Элементы теории нечётких множеств и её приложений / А.П. Рыжов – М.: 2003. – 63 с.
69. Витрик, О.Б. Проблема чувствительной кожи и волоконно-оптические измерительные системы. / О.Б. Витрик // Соросовский образовательный журнал, 2001. – Т. 7, № 1. – С. 108-115.
70. Кульчин, Ю.Н. Датчик температуры на основе волоконно-оптических интерферометров Фабри-Перо с внешними резонаторами / Ю.Н. Кульчин, Ю.Д. Воробьев, О.Б. Витрик, Ю.С. Петров, О.В. Кириченко, О.Т. Каменев, О.Г. Максаев // Оптическая техника, 1997. – Т.12, № 1. – С. 24-25.
71. Кульчин, Ю.Н. Запись голограмм Френеля излучением, прошедшим через многомодовые волоконные световоды / Ю.Н. Кульчин, Ю.А. Быковский, О.Б. Витрик // Оптика и спектроскопия, 1990. – Т. 68, 5. – С. 1160-1169.
72. Кульчин, Ю.Н. Голографическая согласованная фильтрация сигналов в интерференционных датчиках на многомодовых волоконных световодах / Ю.Н. Кульчин, Ю.А. Быковский, О.Б. Витрик, А.И. Ларкин // Квантовая электроника, 1990. – Т. 17, 1. – С. 95-98.
73. Кульчин, Ю.Н. Пространственная фильтрация излучения многомодового световода при измерении гидроакустического давления / Ю.Н. Кульчин, В.Ф. Обух // Квантовая электроника, 1986. – Т. 13. – С. 650.
74. Кульчин, Ю.Н. Применение нейроподобных вычислительных сетей в исследовании физических полей / Ю.Н. Кульчин, О.Б. Витрик, И.В. Денисов, О.Т. Каменев, Р.В. Ромашко // Тезисы докладов IV Всероссийского семинара “Нейроинформатика и ее применения”, Красноярск. 1996. – С. 63.
75. Кульчин, Ю.Н. Самообучающаяся нейронная сеть для обработки томографических данных / Ю.Н. Кульчин, О.Б. Витрик, О.В. Кириченко, Ю.С. Петров, О.Т. Каменев // Сборник трудов “Кибернетика и высшая школа”, 1994. – Т. 28. – С. 3-7.

76. Кульчин, Ю.Н. Оптоэлектронная нейроподобная система обработки выходных данных волоконно-оптической измерительной сети / Ю.Н. Кульчин, И.В. Денисов, О.Т. Каменев // ПЖТФ, 1999. – Т. 25, №6. – С. 65-70.
77. Кульчин, Ю.Н. Волоконно-оптический интерферометрический метод для исследования деформаций строительных конструкций / Ю.Н. Кульчин, О.Б. Витрик, Я.И. Витрик, Л.К. Борисенко, Ю.С. Петров, Е.Г. Шестопапов // Изв. ВУЗов Сер. Строительство, 2001. – Т. 6. – С. 113-117.
78. Kulchin, Yu.N. Reconstruction of Distributed Physical Fields in Integrating Measuring Systems and Systems of Direct Measuring in Rare Points of Field / Yu.N. Kulchin, B.S. Notkin, A.Yu. Kim. // Optical Memory and Neural Networks (Information Optics), 2008. – Vol. 17(2). – P. 93-100.
79. Кульчин, Ю.Н. Волоконно-оптическая система охраны периметра: исследование возможности применения нейронных сетей Ю.Н. Кульчин, Б.С. Ноткин, А.Ю. Ким // Сб. науч. Тр.11-ой Всерос. н.-т. конф. «Нейроинформатика 2009», 2009.–М.: МИФИ, 2009. – т. 2. – С. 259-266.
80. Kulchin, Yu. N. Application of Neural Networks in Fiber-Optics System of Perimeter Defense / Yu.N. Kulchin, B.S. Notkin, A.Yu. Kim, O.T. Kamenev, Yu.S. Petrov // Pacific Science Review, Kangnam University, Republic of Korea, 2010. –Vol. 12(1). – P. 98-101.
81. Kulchin, Yu.N. Data Processing of Fiber-Optics System of Perimeter Defense by means of Neural Networks / Yu.N. Kulchin, A.Yu. Kim. // 20 Anniversary Korea - Russia Science Conference, Baikal, Russia, 2010. – P. 33-35.
82. Наттерер Ф. Математические аспекты компьютерной томографии / Ф. Наттерер – М.: Мир, 1990. – 280 с.
83. Kulchin, Yu.N. Self-training neural network model for real time tomography data processing / Yu.N. Kulchin, O.T. Kamenev // Laser Biology, 1995. – V. 4. – №. 2. – P. 625-629.

84. Кульчин, Ю.Н. Применение персептронов для нелинейной реконструктивной томографии / Ю.Н. Кульчин, И.В. Денисов, А.В. Панов, Н.А. Рыбальченко // Проблемы управления, 2006. - №4. – С. 59-63.
85. Lawson, C.L., Hanson, R.J. Solving Least Squares Problems / C.L. Lawson, R.J. Hanson // Prentice-Hall, 1974, Chapter 23. – P. 161.
86. Хелгасон, С. Преобразование Радона / С. Хелгасон - М.: «Мир», 1983. – 152 с.
87. Deans, S.R. The Radon transform and some of its applications / S.R. Deans – N.Y.: John Wiley and Sons, 1983.
88. Ristic, L. Sensor technology and devices. Chapter one / L. Ristic - Boston: Artech House, 1994. – 520 с.
89. Бусурин, В.И. Оптические и волоконно-оптические датчики / В.И. Бусурин, А.С. Семенов, Н.П. Удалов // Квантовая электроника. – 1985. – Т.12, №5. – С. 901-943.
90. Бусурин, В.И. Волоконно-оптические датчики: Физические основы, вопросы расчёта и применения. / В.И. Бусурин, Ю.Р. Носов – М.: «Энергоатомиздат», 1990. – 255 с.
91. Красюк, Б.А., Световодные датчики / Б.А. Красюк, О.Г. Семенов, А.Г. Шереметьев, В.А. Шестериков – М.: «Машиностроение», 1990. – 256 с.
92. Дмитриев, А.Л. Оптические системы передачи информации: Учебное Пособие / А.Л. Дмитриев – СПб.: СПбГУ ИТМО, 2007. – 96 с.
93. Айбатов, Д.Л. Основы рефлектометрии: Учебное пособие / Д.Л. Айбатов, О.Г. Морозов, Ю.Е. Польский – Казань: ЗАО «Новое знание», 2008. – 116 с.
94. Бутусов, М.М. Волоконная оптика и приборостроение / М.М. Бутусов, С.Л. Галкин, С.П. Оробинский, Б.П. Пал – Л.: «Машиностроение», 1987. – 328 с.
95. Евтихийев, Н.Н. Волоконная и интегральная оптика в информационных системах / Н.Н. Евтихийев, Э.А. Засовин, Д.И. Мировицкий – М.: Изд. МИРЭА. 1987.
96. Снайдер, А. Теория оптических волноводов (Optical Waveruide Theory). Монография / А. Снайдер, Дж. Лав. Перевод: А.Б. Грудинин, В.Б. Сулимов, С.Я.



Фельд, О.Е. Шушпанов. [Перевод с англ. под ред. Е.М. Дианова и В.В. Шевченко] – М.: «Радио и связь», 1987. – 635 с.

97. Моршнеv, С.К. Пропускание светового излучения крутыми изгибами волоконных световодов / С.К. Моршнеv, А.В. Францессон // Квантовая электроника, 1982. - Т.9, №2. – С. 284-291.

98. Аксеноv, В.А. Потери в одномодовых волоконных световодах на однократных изгибах по малому радиусу (Прямоугольный профиль показателя преломления) / В.А. Аксеноv, В.В. Волошин, И.Л. Воробьев, Г.А. Иванов, В.А. Исаев, А.О. Колосовский, С.К. Моршнеv, Ю.К. Чаморовский // Радиотехника и электроника, 2004. - Т.49, №6. – С. 734-742.

99. Аксеноv, В.А. Потери в одномодовых волоконных световодах с треугольным профилем показателя преломления на однократных изгибах по малому радиусу / В.А. Аксеноv, И.Л. Воробьев, Г.А. Иванов, В.А. Исаев, С.К. Моршнеv, Л.П. Прокофьева, Ю.К. Чаморовский, В.В. Щербаков // Радиотехника, 2004. - №12. – С. 27-30.

100. Дейкин, Дус. Оптоволоконные сенсоры: принципы и компоненты / Дус Дейкин, Б. Калш – М.: «Мир», 1992. – 347 с.

101. Ландау, Л.Д. Квантовая механика (нерелятивистская теория). – Издание 2-е, переработанное и дополненное / Л.Д. Ландау, Е.М. Лифшиц — М.: «Наука», 1963. – Т. III. Теоретическая физика. — 704 с.

102. Берестецкий, В.Б. Теоретическая физика / В.Б. Берестецкий, Е.М. Лифшиц, Л.П. Питаевский – М.: «Наука», 1968.— Т. IV. Релятивистская квантовая теория. Часть 1. – 480 с.

103. Лифшиц, Е.М. Теоретическая физика / Е.М. Лифшиц, Л.П. Питаевский — М.: «Наука», 1971.— Т. IV. Релятивистская квантовая теория. Часть 2. – 287 с.

104. Берестецкий, В.Б. Теоретическая физика. — Издание 3-е, исправленное / В.Б. Берестецкий, Е.М. Лифшиц, Л.П. Питаевский — М.: «Наука», 1989. — Т. IV. Квантовая электродинамика. — 720 с.

105. Кульчин, Ю.Н. Метод регистрации деформаций изгиба с применением волоконных световодов с низким значением приведенной частоты / Ю.Н. Кульчин, О.Б. Витрик, А.В. Дышлюк, С.О. Гурбатов // Измерительная техника, 2010. - №2. – С . 47-49.
106. Никоноров Н.В. Материалы и технологии волоконной оптики: специальные оптические волокна. Учебное пособие, курс лекций. / Н.В. Никоноров, А.И. Сидоров – Спб.: СПбГУ ИТМО, 2009. – 130 с.
107. Патлах, А.Л. Светопропускание изогнутых многомодовых оптических волокон / А.Л. Патлах, А.С. Семёнов // Квантовая электроника, 1983. - №7. – С. 868-870.
108. Короленко, П.В. Оптика когерентного излучения: учебное пособие / П.В. Короленко – Москва: Изд-во МГУ им. М.В. Ломоносова, физический факультет, 1997. – 222 с.
109. Глущенко, А.Г. Физические основы волоконной оптики. Конспект лекций / А.Г. Глущенко, М.В. Головкина – Самара: ГОУВПО ПГУТИ, 2009. – 144 с.
110. Шумкова, Д.Б. Специальные волоконные световоды: учеб. пособие. / Д.Б. Шумкова, А.Е. Левченко. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та. - 2011. – 178 с.
111. Агравал, Г. Нелинейная волоконная оптика / Г. Агравал – М.: «Мир», 1996. – 323 с.
112. Базаров, Е.Н. Теоретические основы волоконно-оптической техники / Е.Н. Базаров, В.Д. Бурков, А.Д. Шатров – М.: Изд-во МГУЛ, 2004. – 203 с.
113. Бейли, Д. Волоконная оптика: теория и практика / Д. Бейли, Э. Райт – М.: «Кудиц-Пресс», 2008. – 320 с.
114. Беспалов, В.И. Нелинейная оптика / В.И. Беспалов, Г.А. Пасманик – М.: «Наука», 1980. – 282 с.
115. Беспрозванных, В.Г. Нелинейная оптика: учеб. пособие. / В.Г. Беспрозванных, В.П. Первадчук – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2011. – 200 с.

116. Дианов, Е.М. Волоконная оптика: сорок лет спустя / Е.М. Дианов // Квантовая электроника, 2010. – Т. 40, № 1. – С. 1-6.
117. Дмитриев, В.Г. Прикладная нелинейная оптика / В.Г. Дмитриев, Л.В. Тарасов – М.: «Физматлит», 2004. – 512 с.
118. Желтиков, А.М. Оптика микроструктурированных волокон / А.М. Желтиков – М.: «Наука», 2004.
119. Йоргачев, Д.В. Волоконно-оптические кабели и линии связи / Д.В. Йоргачев, О.В. Бондаренко – М.: «Экотрендз», 2002. – 282 с.
120. Ландсберг, Г.С. Оптика: учеб. пособие для вузов / Г.С. Ландсберг – М.: «Физматлит», 2003. – 848 с.
121. Кульчин, Ю.Н. Исследование модуляции фазы и состояния поляризации в маломодовом волоконном световоде при аксиальных деформациях / Ю.Н. Кульчин, М.И. Беловолов, О.Б. Витрик, Е.М. Дианов, В.Ф. Обух // Квантовая электроника, 1989. – Т. 16, № 11. – С. 2301-2304.
122. Кульчин, Ю.Н. Амплитудная пространственная фильтрация в обработке сигналов одноволоконного многомодового интерферометра / Ю.Н. Кульчин, Ю.А. Быковский, О.Б. Витрик // Квантовая электроника, 1990. – Т. 17, №10. – С. 1377–1378.
123. Кульчин, Ю.Н. Исследование интерферометра сдвига в схеме волоконно-оптического датчика давления / Ю.Н. Кульчин, О.Б. Витрик, В.Ф. Обух, Ю.С. Петров // Измерительная техника, 1992. – Т. 10. – С. 24-26.
124. Кульчин, Ю.Н. Квазираспределенный волоконно-оптический датчик / Ю.Н. Кульчин, О.Б. Витрик, О.В. Кириченко, Ю.С. Петров // Измерительная техника, 1994. – Т. 1. – С. 16-17.
125. Кульчин, Ю.Н. Квазираспределенный волоконно-оптический датчик / Ю.Н. Кульчин, О.Б. Витрик, О.В. Кириченко, Ю.С. Петров, Ю.Д. Воробьев // Измерительная техника, 1993. – Т. 1. – С. 16-17.

126. Окоси, Тамакори. Волоконно-оптические датчики / Тамакори Окоси, К. Окамот, М. Оцу, Х. Нисихара, К. Кюма, К. Хататэ – Л.: «Энергоатомиздат», Ленинградское отделение, 1991. – 256 с.
127. Бусурин, В.И. Оптические и волоконно-оптические датчики / В.И. Бусурин, А.С. Семенов, Н.П. Удалов // Квантовая электроника, 1985. – Т. 12, №5. – С. 901-944.
128. Шереметьев, А.Г. Когерентная волоконно-оптическая связь / А.Г. Шереметьев – М.: «Радио и связь», 1991. – 192 с.
129. Бендат, Дж. Применения корреляционного и спектрального анализа / Дж. Бендат, А. Пирсол – М.: «Мир», 1983. – 312 с.
130. Кульчин, Ю.Н. Метод электронной корреляционной обработки спекл-картин для выделения полезного сигнала одноволоконных многомодовых интерферометров / Ю.Н. Кульчин, О.Б. Витрик, О.Г. Максаев, О.В. Кириченко, О.Т.Каменев // ЖТФ. 1996. – Т. 66, 12. – С. 137-140 .
131. Кульчин, Ю.Н. Адаптивные методы обработки спекл-модулированных оптических полей / Ю.Н. Кульчин, О.Б. Витрик, А.А. Камшилин, Р.В. Ромашко – М.: Физматлит, 2009. – 288 с.
132. Кульчин, Ю.Н. Корреляционная обработка сигналов одноволоконного многомодового интерферометра в режиме возбуждения малого числа мод при измерениях деформационных воздействий / Ю.Н. Кульчин, О.Б. Витрик, А.Д. Ланцов // Квантовая электроника. 2008. – Т. 38, №1. – С. 56-58.
133. Снайдер, А. Теория оптических волноводов / А. Снайдер, Дж. Лав [Пер. с англ. под редакцией Е.М. Дианова и В.В. Шевченко] – М.: Радио и связь, 1987. — 656 с.: ил.
134. Быковский, Ю.А. Коррелированная перестройка спеклов в интерферометре на многомодовом световоде / Ю.А. Быковский, Ю.Н. Кульчин, В.Ф. Обух, В.Л. Смирнов // Квантовая электроника, 1990. – Т. 17, № 8. – С. 561-564.

135. Кульчин, Ю.Н. Статистические характеристики когерентного излучения в многомодовых волоконных световодах / Ю.Н. Кульчин, Ю.А. Быковский, О.Б. Витрик // Известия вузов. Серия Радиофизика, 1990. – Т. 33, 11. – С. 1310-1311.
136. Сергиенко, А.Б. Цифровая обработка сигналов: учеб. пособие. – 3-е изд. / А.Б. Сергиенко – СПб.: БХВ - Петербург, 2011. – 768 с.
137. Гонсалес, Р. Цифровая обработка изображений в среде MATLAB / Р. Гонсалес, Р. Вудс, С. Эддинс – М.: «Техносфера», 2006. – 616 с.
138. Основы волоконной оптической связи. [Перевод с англ. под ред. Дианова, Е.М. Пер. изд.: Barnoski, M.K. Fundamentals of optical fiber communications, USA, 1976] – М.: «Сов. Радио», 1980. – 232 с.
139. Гроднев, И.И. Волоконно-оптические системы передачи и кабели: Справочник / И.И. Гроднев, А.Г. Мурадян, Р.М. Шарафутдинов и др. – М.: «Радио и связь», 1993. – 264 с.
140. Бутусов, М.М. Волоконно-оптические системы передачи: Учебник для вузов / М.М. Бутусов, С.М. Верник, С.Л. Галкин и др. – М.: «Радио и связь», 1992. – 416 с.
141. Убайдуллаев, Р.Р. Волоконно-оптические сети / Р.Р. Убайдуллаев – М.: «Эко-Трендз», 2001. - 267 с.
142. Оленин, Ю.А. Проблемы комплексного обеспечения охранно-территориальной безопасности и физической защиты особо важных объектов Российской Федерации / Ю.А.Оленин // Проблемы объектовой охраны. Сб. научных трудов, Пенза, 2000. – Вып. 1. – С. 8.
143. Введенский, Б.С. Системы охраны периметров на выставке IFSEC'2000 / Б.С. Введенский // Специальная техника, 2000. - №4. – С. 2-9, №5. – С. 2-12.
144. Ларин, А.И. Быстро развертываемые охранные системы / А.И. Ларин // Специальная техника, 2000. - №4. – С. 10-15.
145. Введенский, Б.С. Оборудование для Охраны Периметров / Б.С. Введенский – М.: “Мир безопасности”, 2002. – 112 с.

146. Введенский, Б.С. Подземные датчики для охраны периметров / Б.С. Введенский // Алгоритм безопасности, 2012. - №1. – С. 6-13.
147. Носов, Ю.Р. Оптоэлектроника. – 2е изд., перераб. и доп. / Ю.Р. Носов – М.: «Радио и связь», 1989. – 360 с.
148. Гауэр, Дж. Оптические системы передачи. Пер. с англ. / Дж. Гауэр – М.: «Радио и связь», 1989. – 501 с.
149. Шевцов, Э.А. Фотоприемные устройства волоконно-оптических систем передачи / Э.А. Шевцов, М.Е. Белкин – М.: «Радио и связь», 1992. – 230 с.
150. Заславский, К.Е. Волоконно-оптические системы передачи (ВОСП). Учебное пособие. Часть 1. / К.Е. Заславский – Новосибирск, НЭИС, 1994. – 76 с.
151. Заславский, К.Е. ВОСП. Учебное пособие. Часть 2. / К.Е. Заславский – Новосибирск, СибГАТИ, 1995.– 68 с.
152. Заславский, К.Е. ВОСП. Учебное пособие. Часть 3. / К.Е. Заславский – Новосибирск, СибГАТИ, 1995. – 62 с.
153. Рубичев, Н.А. Измерительные информационные системы. Учебное пособие / Н.А. Рубичев – М. «Дрофа», 2010. – 334 с.: ил.
154. Тельнов, Ю.Ф. Интеллектуальные информационные системы. Учебное пособие / Ю.Ф. Тельнов – М.: Московский международный институт эконометрики, информатики, финансов и права, 2002. – 118 с.
155. Айвазян, С.А. Прикладная статистика. Классификация и снижение размерности / С.А. Айвазян, В.М. Бухштабер, И.С. Енюков, Л.Д. Мешалкин — М.: Финансы и статистика, 1989. – 607 с.
156. Marquardt, D. An Algorithm for Least-Squares Estimation of Nonlinear Parameters / D. Marquardt // SIAM Journal on Applied Mathematics. – 1963. – Vol.11, №2. – P. 431-441.
157. Крянёв, А.В., Луки, Г.В. Математические методы обработки неопределённых данных / А.В. Крянёв, Г.В. Луки – М.: «Физматлит», 2003. – 216 с.
158. Токарев, В.Л. Построение модели по разнотипным данным для информационно-измерительных систем / В.Л. Токарев, И.Н. Набродова //

Управление, вычислительная техника и информационные технологии. Известия ТулГУ. Технические науки, Тула, 2013. – Вып.1. – С. 280-285.

159. Гренандер, У. Лекции по теории образов: Синтез образов. Пер. с англ. И. Гуревича, Т. Дадашева / У. Гренандер [под редакцией Ю. Журавлёва] – М.: «Мир», 1979. – 384 с.

160. Гренандер, У. Лекции по теории образов: Анализ образов. Пер. с англ. И. Гуревича / У. Гренандер [под редакцией Ю. Журавлёва] – М.: «Мир», 1981. – 448 с.

161. Гренандер, У. Лекции по теории образов: Регулярные структуры. Пер. с англ. И. Гуревича / У. Гренандер [под редакцией Ю. Журавлёва] – М.: «Мир», 1983. – 432 с.

162. Вапник, В.Н. Теория распознавания образов (статистические проблемы обучения) / В.Н. Вапник, А.Я. Червоненкис – М.: «Наука». Гл. ред. физ. – мат. лит. – 1974. – 416 с.

163. Ахо, Альфред. В. Построение и анализ вычислительных алгоритмов / Альфред. В. Ахо, Джон.Э. Хопкрофт, Джеффри. Д. Ульман – М.: «Мир», 1979. – 536 с.

164. Ахо, Альфред. В. Структура данных и алгоритмы. Пер. с англ. Учебное пособие / Альфред. В. Ахо, Джон.Э. Хопкрофт, Джеффри.Д. Ульман – М.: Издательский дом «Вильямс», 2000. – 384 с.: ил.

165. Ключарев, А.А. Структуры и алгоритмы обработки данных. Учебное пособие / А.А. Ключарев, В.А. Матяш, С.В. Щекин – СПб.: ГУАП, 2003. – 172 с.

166. Яхьяева, Г.Э. Нечёткие множества и нейронные сети. Учебное пособие / Г.Э. Яхьяева – М.: «Бином» Лаборатория знаний. 2006. – 316 с.: ил.

167. Леоненков, А.В. Нечёткое моделирование в среде MATLAB и fuzzyTECH / А.В. Леоненков – СПб.: БХВ-Петербург, 2003. – 736 с.: ил.

168. Аверкин, А.Н. Нечёткие множества в моделях управления и искусственного интеллекта. / А.Н. Аверкин, И.З. Батыршин, А.Ф. Блишун, В.Б. Силов, В.Б. Тарасо

[Под редакцией Д.А. Поспелова] – М.: «Наука», гл. ред. физ.-мат. лит. 1986. – 312 с. (Проблемы искусственного интеллекта).

169. Прикладные нечёткие системы. [Под редакцией Т. Тэрано, К. Асаи, М. Сугэно; Пер. с япон. канд. техн. наук Ю.Н. Чершышова] – М.: «Мир», 1993. – 368 с., ил.

170. Дюбуа, Д. Теория возможностей. Приложения к представлению знаний в информатике: Пер. с фр. В.Б. Тарасова / Д. Дюбуа, А. Прад [под редакцией С.А. Орловского] – М.: «Радио и связь», 1990. – 288 с.: ил.

171. Кофман, А. Введение в теорию нечётких множеств: Пер. с франц. В. Б. Кузьмина / А. Кофман [под редакцией С.И. Травкина] – М.: «Радио и связь», 1982. – 432 с.: ил.

172. Zadeh, L.A. Fuzzy Sets / L.A. Zadeh // Information and control. 1965. – 8. – P. 338-353.

173. Bellman, R.T. Decision-Making in Fuzzy Environment / R. T. Bellman, L.A. Zadeh // Management Science, 1970. – 17, №4. – P. 141-164.

174. Джеймс, А. Андерсон. Дискретная математика и комбинаторика. Пер. с англ. / А. Андерсон Джеймс – М.: Издательский дом «Вильямс», 2004. – 960 с.: ил.

175. Kulchin, Yu.N. Reconstruction of Distributed Physical Fields in Different Types of Measuring Systems: Comparison and Analysis of Results / Yu.N. Kulchin, B.S. Notkin, A.Yu. Kim // Proceedings of Seventh Asia- Pacific Conference on Fundamental Problems of Opto- and Microelectronics APCOM'2007, г. Владивосток, 2007.

176. Kulchin, Yu.N. Concepts of Intelligent Fiber-Optics System of Perimeter Defense / Yuri N. Kulchin, Boris S. Notkin and Alexandra Yu. Kim // Proceedings of Ninth Asia-Pacific Conference on Fundamental Problems of Opto – and Microelectronics APCOM'2009, г. Владивосток, 2009.

177. Кульчин, Ю.Н. Построение алгоритма нечёткого дерева решений на основе экспериментальных данных при обработке сигналов РВОИС / Ю.Н. Кульчин, А.Ю. Ким, Б.С. Ноткин, А.Б. Люхтер // Информатика и системы управления, 2014. – №3(41). – С. 103-114.



178. Kim, A.Yu. Construction of Fuzzy Decision Tree Based on Experimental Data for Signal Processing of Distributed Fiber-Optic Measuring Network / A.Yu. Kim, Yu.N. Kulchin // Proceedings of Asia- Pacific Conference on Fundamental Problems of Opto – and Microelectronics APCOM'2014, Kokushikan University, Tokyo, 2014. – P. 45-49.
179. Kim, A.Yu. Signal Processing System of Segmental Distributed Fiber-Optic Measuring Network for Recognition of Dynamic Images Using Neural Networks / A.Yu. Kim, Yu.N. Kulchin // Proceedings of Asia-Pacific Conference on Fundamental Problems of Opto – and Microelectronics APCOM'2014, Kokushikan University, Tokyo, 2014. – P. 39-44.
180. Kulchin, Yu.N. Recognition of Dynamic Images by Signal Processing System of Fiber-Optic Measuring Network Using Neural Technology / Yu.N. Kulchin, A.Yu. Kim and B.S. Notkin // Proceeding of Fundamental Science and Technology – Promising Developments IV, North Charleston, USA, 2014. – Vol.1. – P. 183-190.

**СПИСОК СОКРАЩЕНИЙ**

**АЦП** – аналого-цифровой преобразователь;

**ВОД** – волоконно-оптический датчик;

**ВОИС** – волоконно-оптическая измерительная сеть;

**ВОИЛ** – волоконно-оптическая измерительная линия;

**ИИС** – информационно-измерительная система;

**ИИИС** – интеллектуальная информационно-измерительная система;

**ИРИИС** – интеллектуальная распределённая информационно-измерительная система;

**ИС** – измерительный сегмент;

**МНС** – многоканальная нейронная сеть;

**МО** – модель образа;

**НС** – нейронная сеть;

**ОМИ** – одноволоконный многомодовый интерферометр;

**ПНС** – продукционная нечёткая система;

**РВОИС** – распределённая волоконно-оптическая измерительная сеть;

**РИС** – распределённая измерительная система;

**СОД ИРИИС** – система обработки данных ИРИИС;

**ЧЭ** – чувствительный элемент;

**ЭС** – экспертная система.

```

%%          Программа выполнена в соответствии с материалами, изложенными в
%% ===== файле Признаки.doc =====

m=General.NumberLines; % число измерительных линий
P=zeros(length(V),13); % Инициализируем массив для хранения векторов из
                        % 13-ти признаков (этого делать не обязательно,
                        % но правила хорошего тона программирования этого
                        % требуют)

n1=0;
c1=12; % общее количество классов

for n=1:length(V) % последовательно рассматриваем все примеры выборки
                % Формируем массив d, который имеет
                % ту же размерность, что и V(n).Data,
                % но содержит все 0 кроме моментов внешнего воздействия
                % (этим моментам соответствуют 1)
D=[zeros(m,1) V(n).Data]; % готовим исходные данные:
                        % искусственно добавляем в начало один столбец,
                        % что избавляет от введения дополнительных
                        % условий при выполнении строки №31
d=diff([zeros(m,2) D],1,2); % считаем приращение относительно данных,
                        % расширенных двумя нулевыми столбцами
                        % (это позволяет не терять воздействия,
                        % пришедшие уже в первый момент времени)
d(d<0.03)=0; % обнуляем всё, что не превышает шумовой порог
                % (0.03 - уровень порога)
d(d>=0.03)=1; % остальные элементы матрицы d устанавливаем в 1
d=diff(d,1,2); % еще раз считаем приращение
d(d<0)=0; % обнуляем отрицательные приращения
                % В итоге, остаются только единичные,
                % соответствующие моментам приложения внешнего
                % воздействия.
                % Инициализируем переменные, участвующие в преобразовании
                % массива d в массив A, содержащий конкретно
                % информацию о воздействиях.
nLA=0; % обнуляем переменную для хранения числа активных линий
A=[]; % сбрасываем переменную, для хранения массива воздействий
        % (удаляем все данные, оставшиеся после обработки
        % предыдущего примера)
q=0; % обнуляем переменную - счетчик суммарного числа воздействий
StepOnLine=[]; % сбрасываем вектор для хранения числа воздействий
                % по активным изм. линиям формируем массив
                % воздействий A для примера n
for i=1:m % последовательно анализируем все линии
        t=find(d(i,:)); % ищем моменты воздействия на линию номер i
        if ~isempty(t) % если переменная t не пустая
                % (есть воздействия), то:
                nLA=nLA+1; % увеличиваем на 1 счетчик активных линий
                StepOnLine(nLA)=length(t); % запоминаем сколько раз наступили
                                                % на текущую активную линию
                t=[t size(D,2)+1]; % формируем вектор с интервалами времени
                    % для вычисления амплитуды каждого воздействия на линию,
                    % искусственно добавляя к нему номер последнего элемента
                    % (последний отсчет в исходном векторе данных)
                for j=1:length(t)-1 % последовательно рассматриваем все интервалы
                        q=q+1; % увеличиваем счетчик суммарного числа
                                % воздействий на 1
                        A(q,1)=t(j); % в первом столбце сохраняем номер
                                    % дискретного отсчета
                        A(q,2)=i; % во втором столбце сохраняем

```

```

                                % номер изм. линии
A(q,3)=max(D(i,t(j):t(j+1)-1))-D(i,t(j)-1);
                                % в третьем столбце сохраняем амплитуду
                                % воздействия на его интервале времени,
                                % рассчитанную относительно начального сигнала,
                                % сформировавшегося в изм. линии
                                % под воздействием других факторов
    end;
    end;
end;

A=sortrows(A,1); % Сортируем массив воздействий по столбцу времени
                % (для удобства реализации признаков №1 и №5)

A(:,1)=(A(:,1)-A(1,1))*General.SampleTime;
                % пересчитываем дискретное время в реальное.
                % Момент первого воздействия принимаем за 0

A(:,2)=A(:,2)*General.PeriodLines; % Пересчитываем номера линий в метры

% Вид массива A аналогичен предложенному в comment.doc:
% Первый столбец - время;
% Второй столбец - координата;
% Третий столбец - амплитуда.

% ----- Вычисляем значения признаков (описание в Признаки.doc) -----
d=diff(A(:,1)); % считаем приращение времени между соседними воздействиями
d(d==0)=[];
if size(A,1)>1 && ~isempty(d)
    n1=n1+1;
    Mass(n1)=V(n).Mass;
    Speed(n1)=sum(V(n).Speed(1,1):V(n).Speed(end,1))/length(V(n).Speed(:,1));
    E(n1,1)=Mass(n1);
    E(n1,2)=Speed(n1);
    for c=1:c1
        if V(n).NumDescriptor==c
            Class(n1,c)=1;
            E(n1,c+2)=Class(n1,c);
        else
            Class(n1,c)=0;
        end;
    end;
end;

% Признаки (P1-P13):
% 1. Длительность инцидента.
P(n1,1)=A(end,1); % данные отсортированы по возрастанию,
                  % а момент первого воздействия мы уже приняли
                  % за 0 (A(1,1)=0)

% 2. Число активных измерительных линий nLA
P(n1,2)=nLA;

% 3. Среднеарифметическое значение и стандартное отклонение числа
%    воздействий на активных измерительных линиях.
P(n1,3)=sum(StepOnLine)/length(StepOnLine);
P(n1,4)=std(StepOnLine,1);

% 4. Среднеарифметическое значение и стандартное отклонение максимальных
%    амплитуд на активных измерительных линиях
P(n1,5)=sum(A(:,3))/length(A(:,3));
P(n1,6)=std(A(:,3),1);

% 5. Периодичность внешних воздействий на измерительную систему.
d=diff(A(:,1)); % считаем приращение времени между соседними воздействиями
d(d==0)=[];     % удаляем из найденного нулевые приращения
                % (скорее всего, обусловленные "одновременным наступанием"
                % сразу на две изм. линии)

```

```

%     if length(d)==0
%         beep;
%     end;
P(n1,7)= sum(d)/length(d);
P(n1,8)= std(d,1);
% 6. Траектория движения нарушителя (подробное описание содержится в
Признаки.doc)
D=union(A(:,1),A(:,1)); % операция объединения здесь используется для удаления
% повторяющихся элементов в векторе моментов времени A(:,1)
% (они малоинформативны с точки зрения
% аппроксимации степенным полиномом)
if length(D) < 4
    p = length(D)-1;
else
    p = 3;
end;
k = polyfit(A(:,1),A(:,2),p);
% plot(A(:,1), A(:,2), 'ro', [0:0.01:max(A(:,1))],
polyval(k, [0:0.01:max(A(:,1))]))
% если убрать ремарку, то на графике будет изображен только последний
% обработанный пример;
a = k(end:-1:1);
a(5) = 0;
P(n1,9) = a(1);
P(n1,10) = a(2);
P(n1,11) = a(3);
P(n1,12) = a(4);
% 7. Среднеквадратичная ошибка аппроксимации траектории движения нарушителя.
P(n1,13)=mse(A(:,2)-polyval(k, A(:,1)));
end;
end;
S=[sum(Class(:,1)) sum(Class(:,2)) sum(Class(:,3)) sum(Class(:,4))
sum(Class(:,5)) sum(Class(:,6)) sum(Class(:,7)) sum(Class(:,8))
sum(Class(:,9)) sum(Class(:,10)) sum(Class(:,11)) sum(Class(:,12))];
% убираем лишнее из workspace (для анализа)
clear A D StepOnLine a ans d i j k m n nLA p q t c

% -----

%% =====      Обучение и обработка нейронными сетями      =====
close all
clear in1 in2
Set=2; % 1 - включить всё; 2 - включить только скорость или массу; 3 - включить
только класс
Task=1; % 1 - обучение; 2 - обработка; 3 - итог
n2=2; % 1 - масса; 2 - скорость
ad=[1 2 3 4 5 6 7 8 9 10 11 12 13]; % признаки
cls=[1;2;3;4;5;6;7;8]; % классы
% Ref=zeros(length(V)/2, 2);
Ref1=[];
Ref2=[];
k0=1000; % число примеров в тестовой выборке
Ref=zeros(length(V)-k0, 2);
Ref0=zeros(length(E)-(length(V)-k0), 2);
%in1=(P(1:length(V)/2,:))';
in1=(P(1:(length(V)-k0),:))';
%in2=(P(length(V)/2+1:length(V),:))';
in2=(P((length(V)-k0+1):length(E),:))';
switch Set
    case {1,2}
        switch Task % для определения массы и скорости объектов
            case 1 % обучение нейронной сети

```

```

if (n2==1)
    % out1=(E(1:length(V)/2,1')).^(1/3);
    out1=(E(1:(length(V)-k0),1')).^(1/3);
elseif (n2==2)
    % out1=(E(1:length(V)/2,2')).^(1/3);
    out1=(E(1:(length(V)-k0),2')).^(1/3);
end;

Net=newff(minmax(in1(ad,:)),[10 1],{'tansig' 'purelin'}, 'trainlm');
Net.trainParam.epochs=1000;
Net.trainParam.goal=0.001;
Net=train(Net,in1(ad,:),out1);

Y=sim(Net,in1(ad,:)).^3;
if (n2==1)
    Ref1=Ref;
    Ref1(:,1)=Y;
elseif (n2==2)
    Ref2=Ref;
    Ref2(:,2)=Y;
end;

figure
plot(out1.^3,'b');
hold on
plot(sim(Net,in1(ad,:)).^3,'r');

% ошибки обучения:
M1=mse(out1-sim(Net,in1(ad,:)));
M2=max(M1./out1);

case 2 % обработка тестовой выборки примеров обученной нейронной сетью
if (n2==1)
    % out2=(E(length(V)/2+1:length(V),1')).^(1/3);
    out2=(E((length(V)-k0+1):length(E),1')).^(1/3);
elseif (n2==2)
    % out2=(E(length(V)/2+1:length(V),2')).^(1/3);
    out2=(E((length(V)-k0+1):length(E),2')).^(1/3);
end;

Y=sim(Net,in2(ad,:)).^3;
if (n2==1)
    % Ref1=Ref;
    Ref1=Ref0;
    Ref1(:,1)=Y;
elseif (n2==2)
    % Ref2=Ref;
    Ref2=Ref0;
    Ref2(:,2)=Y;
end;

figure
plot(out2.^3,'b');
hold on
plot(sim(Net,in2(ad,:)).^3,'r');

% ошибки обработки тестовой выборки:
MM1=mse(out2-sim(Net,in2(ad,:)));
MM2=max(MM1./out2);

if isempty(Ref1)

```

```

fY=Ref2;
fY2=fY;
elseif isempty(Ref2)
fY=Ref1;
fY1=fY;
end;
    case 3
fY=fY1+fY2;
end;
case 3
end;

switch Set
case {1,3}
% Ref3=zeros(length(V)/2,size(cls,1));
Ref3=zeros((length(V)-k0),size(cls,1));
Ref4=zeros(length(E)-(length(V)-k0),size(cls,1));
R2=Ref3;
% R4=Ref3;
R4=Ref4;
for i=1:size(cls,1)
    switch Task % для распознавания класса
        case 1 % обучение нейронной сети
            % out3=(E(1:length(V)/2,3:cls(i)+2)');
            out3=(E(1:(length(V)-k0),3:cls(i)+2)');

            Net1=newff(minmax(in1(ad,:)),[10 1],{'tansig' 'tansig'}, 'trainlm');
            Net1.trainParam.epochs=1000;
            Net1.trainParam.goal=0.001;
            Net1=train(Net1,in1(ad,:),out3(cls(i),:));

% контроль работы нейронной сети на обучающей выборке:
Y1=sim(Net1,in1(ad,:));
Ref3(:,i+2)=Y1;
R1=Ref3;

A=ones(size(out3(cls(i),:)));
Y3=abs(A-abs(sim(Net1,in1(ad,:))));
R2(:,i)=Y3;
[C0,I0]=min(R2,[],2);
MM0=mse(C0);

figure(i+1)
plot(out3(cls(i),:),'b');
hold on
plot(sim(Net1,in1(ad,:)),'r');

        case 2 % повторное обучение нейронной сети и обработка
            % тестовой выборки примеров обученной нейронной сетью

            % out3=(E(1:length(V)/2,3:cls(i)+2)');
            out3=(E(1:(length(V)-k0),3:cls(i)+2)');

            Net1=newff(minmax(in1(ad,:)),[10 1],{'tansig' 'tansig'}, 'trainlm');
            Net1.trainParam.epochs=1000;
            Net1.trainParam.goal=0.001;
            Net1=train(Net1,in1(ad,:),out3(cls(i),:));

% обработка:
% out4=(E(length(V)/2+1:length(V),3:cls(i)+2)');
out4=(E((length(V)-k0+1):length(E),3:cls(i)+2)');

```

```

Y2=sim(Net1,in2(ad,:));
Ref4(:,i+2)=Y2;
R3=Ref4;

figure(i+1)
plot(out4(cls(i),:),'b');
hold on
plot(sim(Net1,in2(ad,:)),'r');

A=ones(size(out4(cls(i),:)));
Y3=abs(A-abs(sim(Net1,in2(ad,:))));
R4(:,i)=Y3;

% ошибки обработки тестовой выборки:
[C,I]=min(R4,[],2);
d=max(C);
MM=mse(C); % среднеквадратичная ошибка
C1=C;
C1(C<=d)=1;
C1(C>d)=0;
C11=C1.*cls(I,1);
if Set==1
fY0=cat(2,fY,C11);
else
fY0=cat(2,C,C11);
end
case 3
fY01=cat(2,fY,C11); % общий итог (таблица результатов:
% столбцы: 1-масса; 2-скорость; 3-класс)
end;
end;
if Task==2
% корреляционная функция:
out4=out4(cls(1):cls(size(cls)),:);
[R,I1]=max(out4,[],1);
A1=cls(I1).*R';
S1=cat(2,C11,A1);
S2=corrcoef(C11,A1);
end;
case 2
end;

```



**start.m**

```

close all % закрыть все графические окна
N=10;     % номер рассматриваемого примера

u=0.015; % порог
n1=10;   % порядок скользящего среднего для формирования "быстрого" сигнала
n2=20;   % порядок скользящего среднего для формирования "медленного" сигнала

%% графики данных и области воздействия по всем измерительным линиям
figure(1)
t=(0:size(V(N).Data,2)-1)*V(N).SampleTime; % вектор физического времени
for i=1:8
    plot(t,i+V(N).Data(i,:)-1,'-b')
    hold all
    for j=1:length(V(N).Act(i).s)
        plot((V(N).Act(i).s(j) V(N).Act(i).f(j))-1)*V(N).SampleTime,[i+0.1
i+0.1],'-r','LineWidth',2)
    end;
    r=Action_Search(V(N).Data(i,:),n1,n2,u);
    for j=1:size(r,1)
        plot((r(j,1) r(j,2))-1)*V(N).SampleTime,[i-0.1 i-0.1],'-g','LineWidth',2)
    end;

end;
xlabel('Время, с'); ylabel('Номер измерительной линии');

%% Данные для обработки по отдельно взятой линии
nn=2; % Номер линии

u0=u/5;
ma1=movav(V(N).Data(nn,:),n1); % фильтрация скользящим средним порядка n1
ma2=movav(V(N).Data(nn,:),n2); % фильтрация скользящим средним порядка n2
d=ma2-ma1;

figure
subplot(2,1,1)
plot(t,V(N).Data(nn,:))
hold all
plot(t,ma1)
plot(t,ma2)
legend({'Данные', ['Скользящее среднее: ' num2str(n1)], ['Скользящее среднее: '
num2str(n2)]});
grid on

subplot(2,1,2)
plot(t,ma2-ma1)
hold all
plot([0 t(end)],[u u],'k','LineWidth',2)
plot([0 t(end)],[u0 u0],'k','LineWidth',1)
plot([0 t(end)],[0 0],'-k')

```

**priznak\_set.m**

```

%% ----- Сжатие путем выделения признаков -----

% Обучающая выборка:
% P - эталонные входы [признаки/компоненты (првый столбц - время инцидента)]
% Ref - эталонные выходы
%     Ref(:,1) - класс объекта
%     Ref(:,2) - направление движения
%     Ref(:,3) - масса
%     Ref(:,4) - длина пути

close all
P=zeros(length(V),20);
Ref=zeros(length(V),4);
G={'Класс объекта', 'Направление движения', 'Масса', 'Длина пути'};
%% ----- Номера признаков для каждой из подзадач (имена в G) -----
Flags=[14 1;
        11 18;
         5 6;
         7 20];
%% -----

Class=zeros(length(V),1);
Mass=zeros(length(V),1);

s = {'№1: время инцидента'
     '№2: число активных линий'
     '№3: суммарное количество воздействий по всем линиям'
     '№4: среднеквадратичное отклонение числа воздействий'
     '№5: среднеарифметическое значение амплитуд воздействий'
     '№6: среднеквадратичное отклонение амплитуд воздействий'
     '№7: средняя периодичность воздействий'
     '№8: среднеквадратичное отклонение периодичности воздействий'
     '№9: коэффициенты многочлена t^3'
     '№10: коэффициенты многочлена t^2'
     '№11: коэффициенты многочлена t^1'
     '№12: коэффициенты многочлена t^0'
     '№13: среднеквадратичная ошибка аппроксимации траектории движения'
     '№14: непрерывность водействия'
     '№15: параллельность воздействий'
     '№16: одновременность воздействий'
     '№17: длина пути (сумма модулей приращения)'
     '№18: сумма направлений (сумма знаков приращений)'
     '№19: взвешенный центр масс положения (по прямому времени)'
     '№20: взвешенный центр масс положения (по обратному времени)'};

for i=1:length(V)
    P(i,:)=Make_Signs(V(i).Data);
    Ref(i,1)=V(i).Class;
    Ref(i,2)=V(i).Direction;
    Ref(i,3)=V(i).Mass;
    Ref(i,4)=V(i).PathLength;
end;

for i=1:4
    a=minmax(Ref(:,i)');
    C=(Ref(:,i)-a(1))/(a(2)-a(1));
    figure

```

```

subplot(2,2,[1 2])
for j=1:length(C)
    plot(P(j,Flags(i,1)),P(j,Flags(i,2)),'.','Color',[C(j) 0 1-C(j)])
    hold all
end;
xlabel(s{Flags(i,1)});
ylabel(s{Flags(i,2)});
title([G{i} ' (оба класса)'])
grid on

subplot(2,2,3)
a=find(Ref(:,1)==1);
for q=1:length(a)
    j=a(q);
    plot(P(j,Flags(i,1)),P(j,Flags(i,2)),'.','Color',[C(j) 0 1-C(j)])
    hold all
end;
xlabel(s{Flags(i,1)});
ylabel(s{Flags(i,2)});
title([G{i} ' (класс катящихся объектов)'])
grid on

subplot(2,2,4)
a=find(Ref(:,1)==0);
for q=1:length(a)
    j=a(q);
    plot(P(j,Flags(i,1)),P(j,Flags(i,2)),'.','Color',[C(j) 0 1-C(j)])
    hold all
end;
xlabel(s{Flags(i,1)});
ylabel(s{Flags(i,2)});
title([G{i} ' (класс шагающих объектов)'])
grid on
end;

```

### pca\_set.m

```

%% ----- Сжатие методом главных компонент (Principal Component Analysis
(PCA))

% Обучающая выборка:
% P - эталонные входы [признаки/компоненты (правый столбец - время инцидента)]
% Ref - эталонные выходы
%     Ref(:,1) - класс объекта
%     Ref(:,2) - направление движения
%     Ref(:,3) - масса
%     Ref(:,4) - длина пути

close all

Ref=zeros(length(V),4);
G={'Класс объекта', 'Направление движения', 'Масса', 'Длина пути'};

%% ----- Номера признаков для каждой из подзадач (имена в G) -----
Flags=[1 2;
        1 2;
        1 2;
        1 2];

%% -----
Class=zeros(length(V),1);
Mass=zeros(length(V),1);

```

```

s{1}='Время инцидента';
for i=2:size(P,2);
    s{i}=['Компонента №' num2str(i-1)];
end;

P=Make_Signs_PCA(V,20,100);
for i=1:length(V)
    Ref(i,1)=V(i).Class;
    Ref(i,2)=V(i).Direction;
    Ref(i,3)=V(i).Mass;
    Ref(i,4)=V(i).PathLength;
end;

for i=1:4
    a=minmax(Ref(:,i)');
    C=(Ref(:,i)-a(1))/(a(2)-a(1));
    figure
    subplot(2,2,[1 2])
    for j=1:length(C)
        plot(P(j,Flags(i,1)),P(j,Flags(i,2)),'.','Color',[C(j) 0 1-C(j)])
        hold all
    end;
    xlabel(s{Flags(i,1)});
    ylabel(s{Flags(i,2)});
    title([G{i} ' (оба класса)'])
    grid on

    subplot(2,2,3)
    a=find(Ref(:,1)==1);
    for q=1:length(a)

```

### Action\_Search.m

```

function r=Action_Search(data,n1,n2,u)
% data - вектор строка с данными, подлежащими распознаванию
% n1 - порядок скользящего среднего для формирования "быстрого" сигнала
% n2 - порядок скользящего среднего для формирования "медленного" сигнала
% u - порог

u0=u/5;
ma1=movav(data,n1); % фильтрация скользящим средним порядка n1
ma2=movav(data,n2); % фильтрация скользящим средним порядка n2
d=ma2-ma1;
v=find(2*(diff(d>u)>0)); % дискретные моменты времени пересечения порога u
b=find(diff(d>u0)); % дискретные моменты времени пересечения порога u0
r=zeros(length(v),2); % инициализация возвращаемой матрицы
for i=1:length(v) % для всех моментов времени пересечения порога u
    p=find(v(i)>=b(end:-1:1),1); % момент пересечения порога u0 (ближайший слева)
    if isempty(p)
        r(i,:)= [1 b(1)]; % временной диапазон воздействия: начальный участок
    else
        if p<2
            r(i,:)= [b(end) length(data)]; % временной диапазон воздействия:
финальный участок
        else
            r(i,:)= [b(end-p+1) b(end-p+2)]; % временной диапазон воздействия: по
условию >u0
        end;
    end;
end;
end;

```

```
end;
```

```
function r=movav(d,n) % скользящее среднее
% d - вектор строка с данными, подлежащими фильтрации
% n - порядок скользящего среднего
a=cumsum(d);
r=(a(n+1:end)-a(1:end-n))/(n);
r=[a(1:n)./(1:n) r]; % добавка начальных значений с меньшим порядком усреднения
```

### Make\_Signs.m

```
function P=Make_Signs(D)
% Процедура формирования 20-ти признаков по экспериментальным данным
% D - матрицы размера n x d, где n - число измерительных линий, d - данные с
измерительных линий
% P - возвращаемая матрица-строка со значениями 20-ти признаков

%% Параметры процедуры Action_Search для идентификации интервалов воздействия на
изм. линию
u=0.015; % порог
n1=10; % порядок скользящего среднего для формирования "быстрого" сигнала
n2=20; % порядок скользящего среднего для формирования "медленного" сигнала

%% Инициализация переменных
L=size(D,1); % число измерительных линий
P=zeros(1,20); % массив с возвращаемыми признаками
% Для признака №1: длительность инцидента
start_time=Inf; finish_time=-Inf;
% Признак №2: число активных линий
% Признак №3: суммарное количество воздействий по всем линиям
% Признак №4: среднеквадратичное отклонение числа воздействий по линиям
p4=0;
% Признак №5: среднеарифметическое значение амплитуд воздействий по всем линиям
p5=0;
% Признак №6: среднеквадратичное отклонение амплитуд воздействий по всем линиям
p6=0;
% Признак №7: средняя периодичность воздействий
begin_time_set=[]; end_time_set=[];
% Признак №8: среднеквадратичное отклонение периодичности воздействий
% Признаки №№9-12: коэффициенты многочлена, описывающего траекторию движения
объекта
px=[]; py=[];
% Признак №13: среднеквадратичная ошибка аппроксимации траектории движения
% Признак №14: непрерывность воздействия (0...1)
v=zeros(1,size(D,2));
% Признак №15: параллельность воздействий (0..1)
% Признак №16: одновременность воздействий (максимальное число линий с одновременнм
воздействием)
% Признак №17: длина пути (сумма модулей приращения положения)
begin_line_set=[];
% Признак №18: сумма направлений (сумма знаков приращений положения)
% Признак №№19-20: взвешенные центры масс положения (взвешивание прямым и обратным
временем)

%% Подготовительная последовательная обработка данных с измерительных линий
for i=1:L % для всех измерительных линий
Act=Action_Search(D(i,:),n1,n2,u); % получение интервалов времени с
воздействиями на рассматриваемую изм. линию
% Для признака №1
```

```

start_time=min([start_time; Act(:,1)]); % наименьший момент времени
начала воздействия (начало первого воздействия)
finish_time=max([finish_time; Act(:,2)]); % наибольший момент времени конца
воздействия (конец последнего воздействия)
% Признак №2
P(2)=P(2)+~isempty(Act); % если на линии обнаружено
воздействие, то +1 к значению признака
% Признак №3
P(3)=P(3)+size(Act,1); % накапливаемая сумма воздействий
по всем изм. линиям
% Для признака №4
p4=p4+size(Act,1)^2; % накапливаемая сумма квадратов
количеств воздействий по линиям (для вычисления среднеквадратичного отклонения)
% -----
for j=1:size(Act,1) % для всех воздействий по линии
% Для признаков №5 и №6
a=Act(j,1); % время начала воздействия
b=Act(j,2); % время конца воздействия
c=max(D(i,a), D(i,b))-min(D(i,a:b)); % амплитуда воздействия
относительно максимальной амплитуды в моменты времени его начала и конца
p5=p5+c; % накапливаемая сумма амплитуд
воздействий по всем линиям (для вычисления средней амплитуды)
p6=p6+c^2; % накапливаемая сумма квадратов
амплитуд воздействий по всем линиям (для вычисления среднеквадратичного
отклонения)
% Для признаков №9-12
px=[px Act(j,1):Act(j,2)]; % воздействия по координате
времени
py=[py i*ones(1,Act(j,2)-Act(j,1)+1)]; % воздействия по координате
положения (номер изм. линии)
% Для признаков №14-16
v(Act(j,1):Act(j,2))=v(Act(j,1):Act(j,2))+1; % +1 к диапазону значений,
соответствующему интервалу времени j-го воздействия на линию
end
% Для признаков №7 и №8 (№№17-20)
begin_time_set=[begin_time_set; Act(:,1)]; % множество моментов начал
воздействий
end_time_set=[end_time_set; Act(:,2)]; % множество моментов концов
воздействий
% Для признаков №17-20
begin_line_set=[begin_line_set; i*ones(size(Act,1),1)]; % множество номеров
линий с воздействиями (используется в комплексе с begin_time_set)
end;

%% Формирования признаков
% Признак №1
P(1)=finish_time-start_time+1;
% Признак №4
P(4)=sqrt(p4/L-P(3)^2/L^2);
% Признак №5
P(5)=p5/P(3);
% Признак №6
P(6)=sqrt(p6/P(3)-P(5)^2);
% Признак №7
P(7)=(mean(diff(sort(begin_time_set)))+mean(diff(sort(end_time_set))))/2;
% Признак №8
P(8)=std([diff(sort(begin_time_set)); diff(sort(end_time_set))],1);
% Признаки №9-12
P(9:12)=polyfit(px-start_time,py,3);
% Признак №13
P(13)=sqrt(mse(py-polyval(P(9:12),px-start_time)));
% Признак №14

```

```

    P(14)=sum(sign(v(start_time:finish_time)))/P(1);
% Признак №15
    P(15)=sum(v(start_time:finish_time)>1)/P(1);
% Признак №16
    P(16)=max(v(start_time:finish_time));
% Для признаков №№17-20
    a=sortrows([begin_time_set begin_line_set]);
% Признак №17
    P(17)=sum(abs(diff(a(:,2))));
% Признак №18
    P(18)=sum(sign(diff(a(:,2))));
% Признак №19
    P(19)=sum((a(:,1)-start_time).*a(:,2))/sum(a(:,1)-start_time);
% Признак №20
    P(20)=sum((finish_time-a(:,1)).*a(:,2))/sum(finish_time-a(:,1));

```

### Make\_Signs\_PCA.m

```

function P=Make_Signs_PCA(V,h,n)
% V - структура с полями Data (временные ряды с данными) и SampleTime (период
дискретизации)
% h - число возвращаемых главных компонент ( + первым столбцом указывается время
инцидента)
% n - интервал времени, к которому приводится каждый инцидент (от начала первого
воздействия до конца последнего)

%% Параметры процедуры Action_Search для идентификации интервалов воздействия на
изм. линию
u=0.015; % порог
n1=10; % порядок сользящего среднего для формирования "быстрого" сигнала
n2=20; % порядок сользящего среднего для формирования "медленного" сигнала

L=size(V(1).Data,1); % число измерительных линий
A=zeros(1,L*n); % матрица для данных под PCA
Time=zeros(length(V),1);
for z=1:length(V)
    D=V(z).Data;
    st=Inf; ft=-Inf;
    for i=1:L % для всех измерительных
линий
        Act=Action_Search(D(i,:),n1,n2,u); % получение интервалов времени
с воздействиями на рассматриваемую изм. линию
        st=min([st; Act(:,1)]); % наименьший момент времени
начала воздействия (начало первого воздействия)
        ft=max([ft; Act(:,2)]); % наибольший момент времени
конца воздействия (конец последнего воздействия)
    end;
    Time(z)=(ft-st+1)*V(z).SampleTime;
    for i=1:L % для всех измерительных
линий
        A(z,(i-1)*n+1:i*n)=interp1(st:ft, D(i,st:ft), linspace(st,ft,n)); %
приведение каждого инцидента к единой продолжительности
    end;
end;
H=princomp(A);
P=[Time A*H(:,1:h)];

```

```

%% ===== Задача и способ сжатия данных =====
Task=4;      % 1 - класс; 2 - направление движения; 3 - масса; 4 - длина пути
Compress=1;  % 1 - сжатие признаками, 2 - сжатие главными компонентами
%% -----

if Compress==1
    load('priznak_set_10200.mat');
else
    load('pca_set_10200.mat');
end;

switch Task
    case 1 % Распознавание класса
        if Compress==1
            % Для сжатия признаками
            ad=[7 14]; % Признаки
            net=newff(P(ad,:),T(Task,:),[3 1]); % структура ИНС
        else
            % Для сжатия главными компонентами (список номеров компонент (номер 1
- это время инцидента (компоненты начинаются с номера 2))
            % Варианты двухэлементных входных векторов:
            % ad=1:2;
            % ad=2:3;

            % Надежный вариант:
            ad=1:3;

            net=newff(P(ad,:),T(Task,:),[5 1]);
        end;

    case 2 % Направление движения
        if Compress==1
            % Для сжатия признаками
            ad=[19 20]; % Признаки
            net=newff(P(ad,:),T(Task,:),[1]); % структура ИНС
        else
            % Для сжатия главными компонентами (список номеров компонент (номер 1
- это время инцидента (компоненты начинаются с номера 2))
            % Варианты двухэлементных входных векторов:
            % ad=1:2;
            % ad=2:3;

            % Надежный вариант:
            ad=1:5;
            net=newff(P(ad,:),T(Task,:),[10 1]);
        end;

    case 3 % Масса
        if Compress==1
            % Для сжатия признаками
            % Двухэлементные
            ad=[5 14];
            % трех- и более элементные
            %ad=[3 5 15];
            %ad=[3 5 6 15];
            %ad=[3 5 6 15 20];
            % ad=[1 5 6 11 14 15];
            net=newff(P(ad,:),T(Task,:),[10]); % структура ИНС
        else

```



```

        % Для сжатия главными компонентами
        ad=1:10; % список номеров компонент (номер 1 - это время инцидента
(компоненты начинаются с номера 2))
        net=newff(P(ad,:),T(Task,:),[10]);
        end;

    case 4 % Длина пути
        if Compress==1
            % Для сжатия признаками
            %ad=[3 15];
            %ad=[3 5 15];
            %ad=[3 6 15 19];
            %ad=[3 5 6 15 19];
            %ad=[3 4 5 6 15 19];
            %ad=[3 5 6 8 10 15 20];
            %ad=[3 5 6 8 10 12 15 20];

            net=newff(P(ad,:),T(Task,:),[10 5]); % структура ИНС
        else
            % Для сжатия главными компонентами
            ad=1:20; % список номеров компонент (номер 1 - это время инцидента
(компоненты начинаются с номера 2))
            net=newff(P(ad,:),T(Task,:),[10 5]);
        end;
    end;

end;

close all
Tr=201:10200;
Val=1:200;
net.divideFcn = 'divideind';
net.divideParam.trainInd=Tr;
%net.divideParam.valInd=Val;
net.divideParam.testInd=Val;

s = {'№1: дискретное время продолжительности инцидента'
    '№2: число активных линий'
    '№3: суммарное количество воздействий по всем линиям'
    '№4: среднеквадратичное отклонение числа воздействий'
    '№5: среднеарифметическое значение амплитуд воздействий'
    '№6: среднеквадратичное отклонение амплитуд воздействий'
    '№7: средняя периодичность воздействий'
    '№8: среднеквадратичное отклонение периодичности воздействий'
    '№9: коэффициенты многочлена t^3'
    '№10: коэффициенты многочлена t^2'
    '№11: коэффициенты многочлена t^1'
    '№12: коэффициенты многочлена t^0'
    '№13: среднеквадратичная ошибка аппроксимации траектории движения'
    '№14: непрерывность воздействия'
    '№15: параллельность воздействий'
    '№16: одновременность воздействий'
    '№17: длина пути (сумма модулей приращения)'
    '№18: сумма направлений (сумма знаков приращений)'
    '№19: взвешенный центр масс положения (по прямому времени)'
    '№20: взвешенный центр масс положения (по обратному времени)'};
if length(ad)==2
    if Compress==1
        LX=s{ad(1)};
        LY=s{ad(2)};
    else

```

```

if ad(1)>1
    LX=['Компонента №' num2str(ad(1)-1)];
else
    LX='Время продолжительности инцидента, с';
end;
if ad(2)>1
    LY=['Компонента №' num2str(ad(2)-1)];
else
    LY='Время продолжительности инцидента, с';
end;
end;
end;

```

```
switch Task
```

```

case 1 %%----- Распознавание класса-----
    net=trainnn(net,5);
    disp('Класс');
    disp(['Ошибка по обучающему множеству: ' num2str(100*sum(abs(T(Task,Tr)-
(sim(net,P(ad,Tr))>0.5)))/length(Tr)) '%;'])
    disp(['Ошибка по тестовому множеству: ' num2str(100*sum(abs(T(Task,Val)-
(sim(net,P(ad,Val))>0.5)))/length(Val)) '%;'])
    if length(ad)==2
        figure
        p1min=min(P(ad(1),:));
        p1max=max(P(ad(1),:));
        p2min=min(P(ad(2),:));
        p2max=max(P(ad(2),:));
        nn=500;
        [P1 P2]=meshgrid(linspace(p1min,p1max,nn),linspace(p2min,p2max,nn));
        Z=sim(net,[P1(:) P2(:)'])>0.5;
        contourf(P1,P2,reshape(Z,nn,nn),1,'LineWidth',2)
        %alpha(0.9)
        C=[1 1 0.6; 0.6 1 0.6];
        colormap(C);
        hold all
        for i=Tr
            if T(Task,i)==1
                plot(P(ad(1),i),P(ad(2),i),'xk')
            else
                plot(P(ad(1),i),P(ad(2),i),'.k')
            end;
        end;
        for i=Val
            if T(Task,i)==1
                plot(P(ad(1),i),P(ad(2),i),'xr','LineWidth',2)
            else
                plot(P(ad(1),i),P(ad(2),i),'ob','LineWidth',2)
            end;
        end;
        xlabel(LX);
        ylabel(LY);
    else
        plot(T(Task,:), 'LineWidth',2)
        hold all
        plot(sim(net,P(ad,:)))
    end;
end;

```

```

case 2 %% Случай №2 ----- Направление -----
---
    net=trainnn(net,5);

```

```

disp('Направление движения');
disp(['Ошибка по обучающему множеству: ' num2str(100*sum(abs(T(Task,Tr)-
(sim(net,P(ad,Tr))>0.5))/length(Tr)) '%;'])
disp(['Ошибка по тестовому множеству: ' num2str(100*sum(abs(T(Task,Val)-
(sim(net,P(ad,Val))>0.5))/length(Val)) '%;'])
if length(ad)==2
    figure
    p1min=min(P(ad(1),:));
    p1max=max(P(ad(1),:));
    p2min=min(P(ad(2),:));
    p2max=max(P(ad(2),:));
    nn=500;
    %[P1 P2]=meshgrid(linspace(p1min-(p1max-p1min)*0.1,p1max+(p1max-
p1min)*0.1,nn),linspace(p2min-(p2max-p2min)*0.1,p2max+(p2max-p2min)*0.1,nn));
    [P1 P2]=meshgrid(linspace(p1min,p1max,nn),linspace(p2min,p2max,nn));
    Z=sim(net,[P1(:) P2(:)'])>0.5;
    contourf(P1,P2,reshape(Z,nn,nn),1)
    C=[1 1 0.6; 0.6 1 0.6];
    colormap(C);
    hold all
    for i=Tr
        if T(Task,i)==1
            plot(P(ad(1),i),P(ad(2),i),'xk')
        else
            plot(P(ad(1),i),P(ad(2),i),'.k')
        end;
    end;
    for i=Val
        if T(Task,i)==1
            plot(P(ad(1),i),P(ad(2),i),'xr','LineWidth',2)
        else
            plot(P(ad(1),i),P(ad(2),i),'ob','LineWidth',2)
        end;
    end;
    xlabel(LX);
    ylabel(LY);
else
    plot(T(Task,:), 'LineWidth',2)
    hold all
    plot(sim(net,P(ad,:)))
end;

```

```

case 3 %% Случай №3 ----- Масса -----
mm=minmax(T(Task,:));
net=trainn(net,5);
disp('Масса');
disp(['Ошибка по обучающему множеству: ' num2str(100*sum(abs(T(Task,Tr)-
sim(net,P(ad,Tr)))/T(Task,Tr))/length(Tr)) '%;'])
disp(['Ошибка по тестовому множеству: ' num2str(100*sum(abs(T(Task,Val)-
sim(net,P(ad,Val)))/T(Task,Val))/length(Val)) '%;'])
if length(ad)==2
    figure
    p1min=min(P(ad(1),:));
    p1max=max(P(ad(1),:));
    p2min=min(P(ad(2),:));
    p2max=max(P(ad(2),:));
    nn=500;
    %[P1 P2]=meshgrid(linspace(p1min-(p1max-p1min)*0.1,p1max+(p1max-
p1min)*0.1,nn),linspace(p2min-(p2max-p2min)*0.1,p2max+(p2max-p2min)*0.1,nn));
    [P1 P2]=meshgrid(linspace(p1min,p1max,nn),linspace(p2min,p2max,nn));
    Z=sim(net,[P1(:) P2(:)']);

```

```

h=contourf(P1,P2,reshape(Z,nn,nn),20,'LineWidth',1);
C=zeros(size(jet));
i=0;
for c=linspace(0,1,64);
    i=i+1;
    C(i,:)=[c 0 1-c];
end;
colormap(C)
alpha(0.2)

hold all
for i=Tr
    c=(T(Task,i)-mm(1))/(mm(2)-mm(1));
    plot(P(ad(1),i),P(ad(2),i),'.','Color',[c 0 1-c])
end;
for i=Val
    c=(T(Task,i)-mm(1))/(mm(2)-mm(1));
    plot(P(ad(1),i),P(ad(2),i),'x','Color',[c 0 1-c],'LineWidth',3)
end;
xlabel(LX);
ylabel(LY);
else
    plot(T(Task,:), 'LineWidth',2)
    hold all
    plot(sim(net,P(ad,:)))
end;

case 4 %% Случай №4 ----- Длина пути -----
--
mm=minmax(T(Task,:));
net=trainn(net,5);
disp('Длина пути');
disp(['Ошибка по обучающему множеству: ' num2str(100*sum(abs(T(Task,Tr)-
sim(net,P(ad,Tr)))/T(Task,Tr))/length(Tr)) '%;'])
disp(['Ошибка по тестовому множеству: ' num2str(100*sum(abs(T(Task,Val)-
sim(net,P(ad,Val)))/T(Task,Val))/length(Val)) '%;'])
if length(ad)==2
    figure
    p1min=min(P(ad(1),:));
    p1max=max(P(ad(1),:));
    p2min=min(P(ad(2),:));
    p2max=max(P(ad(2),:));
    nn=500;
    %[P1 P2]=meshgrid(linspace(p1min-(p1max-p1min)*0.1,p1max+(p1max-
p1min)*0.1,nn),linspace(p2min-(p2max-p2min)*0.1,p2max+(p2max-p2min)*0.1,nn));
    [P1 P2]=meshgrid(linspace(p1min,p1max,nn),linspace(p2min,p2max,nn));
    Z=sim(net,[P1(:) P2(:)]');
    Z(Z<mm(1))=mm(1);
    Z(Z>mm(2))=mm(2);
    h=contourf(P1,P2,reshape(Z,nn,nn),20,'LineWidth',1);
    C=zeros(size(jet));
    i=0;
    for c=linspace(0,1,64);
        i=i+1;
        C(i,:)=[c 0 1-c];
    end;
    colormap(C)
    alpha(0.1)

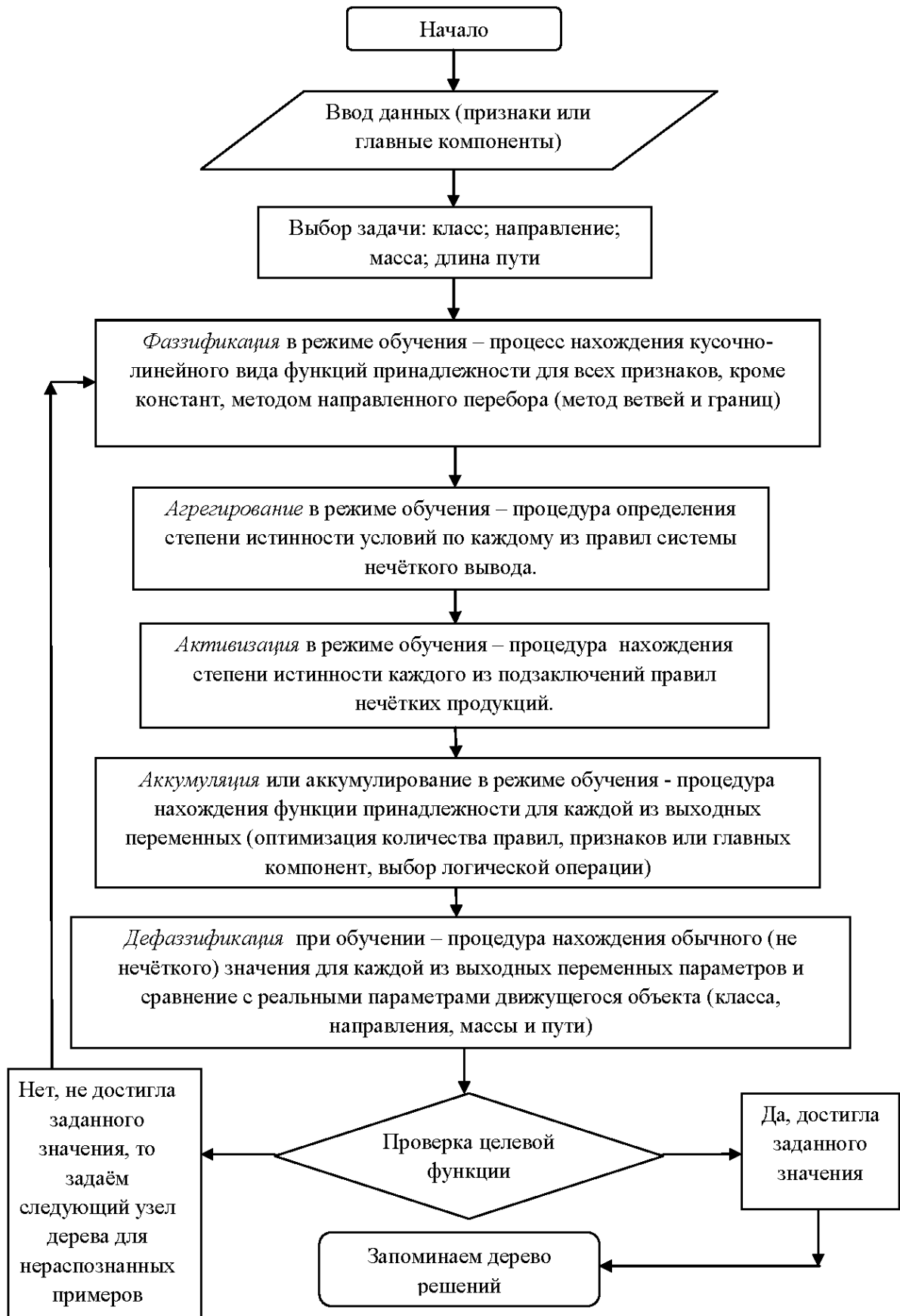
    hold all
    for i=Tr
        c=(T(Task,i)-mm(1))/(mm(2)-mm(1));

```

```
        plot(P(ad(1),i),P(ad(2),i),'.','Color',[c 0 1-c])
    end;
    for i=Val
        c=(T(Task,i)-mm(1))/(mm(2)-mm(1));
        plot(P(ad(1),i),P(ad(2),i),'x','Color',[c 0 1-c],'LineWidth',3)
    end;
    xlabel(LX);
    ylabel(LY);
else
    plot(T(Task,:), 'LineWidth',2)
    hold all
    plot(sim(net,P(ad,:)))
end;

end;
```

## Блок-схема алгоритма построения нечёткого дерева решений



```

%% Очистить все
clear all
close all
clc

%% Загрузка данных
% Имеются две базы данных, каждая из которых содержит 10200 примеров (200 в начале
оригинальные, остальные сгенерированы)
% признак_set_10200.mat - данные для обучения по признакам (20 подготовлены
ручками признаков)
% pca_set_10200.mat - данные для обучения по главным компонентам (Время инцидента
+ 20 главных компонент = 21 шт.)
% Каждый файл содержит по четыре переменных:
% P - входы
% T - выходы (в строках: 'Класс объекта', 'Направление движения', 'Масса, кг',
'Длина пути, м')
% PName - названия признаков (не влияют на исход решения, нужны для понимания
того, что происходит)
% TName - обозначения задач на выходе (присутствуют, чтобы не забыть)

%load('признак_set_10200.mat'); % признаки на входе
load('pca_set_10200.mat'); % главные компоненты на входе

%% ----- Выбор целевого параметра, Подготовка данных для обучения и
тестирования -----
% обучающая выборка в матрицах: Factor (входы) и Target (выход)
% тестовая выборка в матрицах: Test_Factor (входы) и Test_Target (выход)

%% ===== Выбор задачи =====
p=1; % 1- 'Класс объекта'; 2 - 'Направление движения'; 3 - 'Масса, кг'; 4 -
'Длина пути, м'

TrainSamples=201:10200; % порядковые номера примеров для обучения
TestSamples=1:200; % порядковые номера примеров для тестирования
Attr=1:size(P,1); % порядковые номера учитываемых признаков на
входе!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

Factor = P(Attr, TrainSamples);
Test_Factor = P(Attr, TestSamples); % для всех 4 задач входы общие,
% а вот выходной вектор формируется индивидуально для каждой задачи

switch p
    case {1, 2}
        Target = T(p, TrainSamples);
        Test_Target = T(p, TestSamples);
    case {3, 4}
        if p==3 % Масса
            u=[-Inf 2 3.25 8 Inf]; interval=1; % Задание и выбор интервала
от u(i) до u(i+1) для массы
        else % Длина пути
            u=[-Inf 0.63 0.8 Inf]; interval=1; % Задание и выбор интервала от
u(i) до u(i+1) для длины пути (путь / время = средняя скорость [м/с])
        end;
end;

```

```

        Target = (T(p, TrainSamples)>u(interval) & T(p,
TrainSamples)<=u(interval+1));
        Test_Target = (T(p, TestSamples)>u(interval) & T(p,
TestSamples)<=u(interval+1));
end;

%% Обучение
Membership_Function_Intervals = 20;
Step = 0.05;
Allowable_Error = 50;
Names = PName(Attr);

F=fdt(Factor, Target, Membership_Function_Intervals, Step, Allowable_Error,
Names);

% Factor - эталонные входы
% Target - эталонные выходы
% Membership_Function_Intervals - кусочно-линейных интервалов для представления
функции принадлежности
% Step - шаг ординаты функции принадлежности (используется при оптимизационном
переборе)
% Allowable_Error - допустимое число ошибочно классифицированных примеров в
подмножестве
% Names - имена входных признаков (отображаются при визуализации функций
принадлежности)

%% Оценка качества классификации:
% по обучающей выборке
Ans=fdt_solve(F, Factor); % ответ нечеткого дерева F на вход Factor
Err=sum(((Ans>0.5)-Target)~=0); % суммарное число ошибочных ответов (Target -
эталонные ответы)
disp(['Ошибок по обучающей выборке: ' num2str(Err) ' из ' num2str(length(Target))
' - ' num2str(100*Err/length(Target)) '%'])

% по тестовой выборке
Ans=fdt_solve(F, Test_Factor); % ответ нечеткого дерева F на вход
Test_Factor
Err=sum(((Ans>0.5)-Test_Target)~=0); % суммарное число ошибочных ответов
(Test_Target - эталонные ответы)
disp(['Ошибок по тестовой выборке: ' num2str(Err) ' из '
num2str(length(Test_Target)) ' - ' num2str(100*Err/length(Test_Target)) '%'])

%% Показать структуру дерева
% fdt_show(F) % Показать на примере данных обучения
% fdt_solve(F, Test_Factor, Test_Target); % Показать на примере тестовых данных

%% Удаляем временные переменные
clear Ans Err Factor Target TestSamples TrainSamples interval p u

```

### SolveLogic.m

```

function Ans=SolveLogic(G, LC)
% функция SolveLogic используется пользовательскими функциями fdt и fdt_solve
% G - матрица со значениями функций принадлежности
% LC - логическая операция {'min', 'max', 'prod' или 'sum'}
% Ans - результат нечеткой операции

n=size(G,1);

```



```

if n>1
    % Поддерживаемые операции над нечеткими множествами (смотреть, например, тут:
    http://dmtsoft.ru/bn/476/as/oneaticleshablon/):
    switch LC
        case 'min'          % пересечение
            Ans=min(G);
        case 'max'          % объединение
            Ans=max(G);
        case 'prod'         % алгебраическое произведение
            Ans=prod(G);
        case 'sum'          % алгебраическая сумма (рекуррентно применяется к каждому
    последующему аргументу нечеткой операции)
            Ans=G(1,:);
            for i=2:n
                Ans=Ans+G(i,:)-Ans.*G(i,:);
            end;
    end;
else
    Ans=G;
end;

```

### fdt.m

```

function F=fdt(Factor, Target, Membership_Function_Intervals, Step,
Allowable_Error, varargin)
% FDT -Fuzzy Decision Tree
% функция построения нечеткого дерева
%
% Синтаксис:
% F=fdt(Factor, Target, Membership_Function_Intervals, Step, Allowable_Error)
% F=fdt(Factor, Target, Membership_Function_Intervals, Step, Allowable_Error,
Names)

% Factor - эталонные входы
% Target - эталонные выходы
% Membership_Function_Intervals - число кусочно-линейных интервалов для
представления функции принадлежности
% Step - шаг по ординате функции принадлежности (используется при оптимизационном
переборе)
% Allowable_Error - допустимое число ошибочно классифицированных примеров в
подмножестве
% Names - имена входных признаков (отображаются при визуализации функций
принадлежности, если параметр Names не задан, то имена назначаются автоматически)

close all

%% Инициализация
if ~isempty(varargin)
    Names=varargin{1};
else
    for i=1:size(Factor,1);
        Names{i}=['Признак №' num2str(i)];
    end;
end;

LogicConstructions={'min', 'max', 'prod', 'sum'};

%% Подготовка переменных
F=[]; % Итоговая структура с решением
clear NB
NB{1}=1:size(Factor,2); % Список номеров примеров для каждого узла дерева
(length(F)==length(NB))

```



```

                                G(j, H(j).I{q}) =
interp1(H(j).Grid(H(j).NodeXY(q:q+2,1)), H(j).MF(i).Data(H(j).NodeXY(q:q+2,2)),
Factor(j, H(j).I{q})); % итерационно корректируем функцию принадлежности
                                Ans=SolveLogic(G(Adr,H(j).I{q}),
LogicConstructions{i}); % Нечеткий логический вывод (в соответствии с текущей
логической операцией LogicConstructions{i})
                                e=Error_Fun(Target(H(j).I{q}), Ans, TW); %
значение целевой функции

                                if e<e_min
                                    e_min=e;
                                    w_best=w; % запоминаем лучшее
значение функции принадлежности в рассматриваемом узле (кусочно-линейная
структура)

                                end;
                                end;
                                H(j).MF(i).Data(q)=w_best;
                                G(j,
H(j).I{q})=interp1(H(j).Grid(H(j).NodeXY(q:q+2,1)),
H(j).MF(i).Data(H(j).NodeXY(q:q+2,2)), Factor(j, H(j).I{q}));
                                end;
                                end;
                                if ishandle(hWB)
                                    waitbar(j/length(H),hWB)
                                else
                                    break
                                end
                                end;
                                Ans=SolveLogic(G(Adr,NB{cb}), LogicConstructions{i}); %
Нечеткий логический вывод
                                Err=Error_Fun(Target(NB{cb}), Ans, TW); %
Значение целевой функции
                                end;
                                if Err>EOld+Allowable_Error
                                    StructOptOk=true;
                                else
                                    %% Запоминаем лучшее к настоящему времени
                                    Temporal_Save(i).Adr = Adr;
                                    Temporal_Save(i).TrainError=Err;
                                    Temporal_Save(i).x=[];
                                    Temporal_Save(i).y=[];
                                    Temporal_Save(i).Name=[];

                                    Temporal_Save(i).Next=zeros(1,2); % номера следующих узлов для
веток дерева (ответов) 0 и 1
                                    Temporal_Save(i).TrainSamples = length(NB{cb});
                                    Temporal_Save(i).True=0;
                                    Temporal_Save(i).False=0;
                                    Temporal_Save(i).NumTrueErr=0;
                                    Temporal_Save(i).NumFalseErr=0;

                                    Temporal_Save(i).LogicOperation = LogicConstructions{i};

                                    for j=1:length(Adr)
                                        Temporal_Save(i).Name{j}=Names{Adr(j)};
                                        Temporal_Save(i).x{j} = H(Adr(j)).Grid(H(Adr(j)).NodeXY(:,1));
                                        Temporal_Save(i).y{j} =
H(Adr(j)).MF(i).Data(H(Adr(j)).NodeXY(:,2));
                                    end;
                                    Temporal_Save(i).Deposit=[];

                                    %% Если осталось еще то, что можно оптимизировать

```

```

if length(Adr)~=1
    EOld=Err;

    %% последовательно выкидываем по одному признаку и фиксируем
ошибку
    D=zeros(length(Adr),2);
    for j=1:length(Adr)
        Adr0=Adr;
        Adr0(j)=[];
        Ans=SolveLogic(G(Adr0,NB{cb}), LogicConstructions{i});
        D(j,1)=Error_Fun(Target(NB{cb}), Ans, TW);
        D(j,2)=std(H(j).MF(i).Data,1);
    end;
    Temporal_Save(i).Deposit=D(:,1);

    %% Исключаем бесполезные признаки (с проверкой на дубли,
    позволяющие безболезненно выкинуть любой экземпляр из пары, но не всю пару)
    [q w]=sortrows(D);
    r=find(cumsum(q(:,1)-Err)>Allowable_Error,1);
    if isempty(r)
        r=size(D,1);
    end;
    etmp=Inf;
    while r>1 && etmp-Err>Allowable_Error
        r=r-1;
        Ans=SolveLogic(G(Adr(w(r+1:end))),NB{cb}),
LogicConstructions{i});
        etmp=Error_Fun(Target(NB{cb}), Ans, TW);
    end;

    %% Вносим изменения в структуру H с описанием функций
    принадлежности для признаков и переходим к следующей итерации
    for j=1:r
        H(Adr(w(j))).MF(i).Act=false;
    end;
    Adr(w(1:r))=[];
    Ans=SolveLogic(G(Adr,NB{cb}), LogicConstructions{i});
    Err=Error_Fun(Target(NB{cb}), Ans, TW);
    E_last=-1;
else
    StructOptOk=true;
end;

end;
end;
if ~ishandle(hWB)
    break
end

end;

%% Выбор и фиксация лучшей логической операции в рассматриваемом узле дерева
решений
if ishandle(hWB)
    minE=Temporal_Save(1).TrainError;
    n_minE=1;
    for i=2:length(LogicConstructions)
        if Temporal_Save(i).TrainError<minE
            minE=Temporal_Save(i).TrainError;
            n_minE=i;
        end;
    end;
end;

```

```

end;
F{cb}=Temporal_Save(n_minE);

%% Нечеткий логический вывод
FazzyAns=FazzyNodeAns( Factor(Temporal_Save(n_minE).Adr, NB{cb}),
Temporal_Save(n_minE).x, Temporal_Save(n_minE).y, LogicConstructions{n_minE});

%% анализ ответов
a=FazzyAns>0.5;
F{cb}.False=sum(~a);
F{cb}.NumFalseErr=sum(Target(NB{cb} (~a))>0.5);
if F{cb}.NumFalseErr>Allowable_Error % кол-во ошибок по ответу:
логический 0
    NB{length(NB)+1}=NB{cb} (~a);
    F{cb}.Next(1)=length(NB);
end;
F{cb}.True=sum(a);
F{cb}.NumTrueErr=sum(Target(NB{cb} (a))<=0.5);
if F{cb}.NumTrueErr>Allowable_Error % кол-во ошибок по ответу: логическая
1
    NB{length(NB)+1}=NB{cb} (a);
    F{cb}.Next(2)=length(NB);
end;
end;
end;

if ishandle(hWB)
    close(hWB);
    fdt_show(F)
end;

function H=InitMarking(P,N,Attr,FGrid,LogicConstructions)
% P - массив значений признаков [признак x номер примера]
% N - индексы рассматриваемых примеров
% Attr - индексы рассматриваемых (активных) признаков
% n - интервалов на функции принадлежности (кусочно-линейная функция)

for i=1:size(P,1)
    for j=1:length(LogicConstructions)
        H(i).MF(j).Act=false;
    end;
end;

for j=1:length(Attr)
    attr=Attr(j);

    h = find(histc(P(attr,N), FGrid{attr})~=0);
    a=unique([FGrid{attr}(h) FGrid{attr}(h+1)]);
    n=length(a);
    H(attr).NodeXY=[(1:n+2)' [1; (1:n)'; n]];
    H(attr).Grid=[2*FGrid{attr}(1)-FGrid{attr}(2) a 2*FGrid{attr}(end)-
FGrid{attr}(end-1)];

    for i=1:length(LogicConstructions)
        H(attr).MF(i).Act=true;
        switch LogicConstructions{i}
            case 'min'
                H(attr).MF(i).Data=ones(1,n);
            case 'max'
                H(attr).MF(i).Data=zeros(1,n);
        end
    end
end

```

```

        case 'prod'
            H(attr).MF(i).Data=ones(1,n);
        case 'sum'
            H(attr).MF(i).Data=zeros(1,n);
        end;
    end;
    end;
    [s ind]=sort(P(attr,N));

    b=zeros(1,n);
    b(1)=1;
    for i=1:n-2
        b(i+1)=find(s(b(i)+1:end)>=a(i+1),1)+b(i)-1;
    end;
    b(end)=length(s);

    H(attr).I{1}=N(ind(b(1):b(2)));
    for i=2:n-1
        if b(i+1)-b(i-1)>0
            H(attr).I{i}=N(ind(b(i-1)+1:b(i+1)));
        else
            H(attr).I{i}=[];
        end;
    end;
    H(attr).I{n+1}=N(ind(b(n-1)+1:b(n)));
end;

function TW=TargetWeight(Target,N)
% TW(1) - вес нуля
% TW(2) - вес единицы

a1=sum(Target(N)>0.5);
a0=length(N)-a1;

if a1~=0 && a0~=0
    TW(1)=1/a0;
    TW(2)=1/a1;
    TW=TW/(a1*TW(2)+a0*TW(1))*length(N);
elseif a1==0
    TW(1)=1;
    TW(2)=0;
else
    TW(1)=0;
    TW(2)=1;
end;

function [G Adr]=SolveFazzy(P,H,n)
% вычисляем значения функций принадлежности для примеров с индексами N (G),
формируем список активных признаков (Adr)
% P - массив признаков
% H - разметка функций принадлежности и их вид
% n - порядковый номер логической операции в векторе LogicConstructions={'min',
'max', 'prod', 'sum'};

G=zeros(size(P));
Adr=zeros(1,size(P,1));
j=0;
for i=1:length(H)
    if H(i).MF(n).Act

```

```

        j=j+1;
        Adr(j)=i;
        G(i,:)=interp1(H(i).Grid(H(i).NodeXY(:,1)),
H(i).MF(n).Data(H(i).NodeXY(:,2)),P(i,:));
        end;
    end;
    Adr(j+1:size(P,1))=[];

```

```

function E=Error_Fun(Target, Current, TW)
a=Target>0;
k=ones(size(Target));
k(a)=TW(2);
k(~a)=TW(1);
E=sum(abs((Current>0.5)-Target).*k)+sum(abs(Target-Current).*k)/length(Target);

```

```

function r=FazzyNodeAns(P, x, y, LC)
G=zeros(size(P));
for i=1:size(P,1)
    G(i,:)=interp1(x{i}, y{i}, P(i,:));
end;
r=SolveLogic(G, LC);

```