МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

«Тверской государственный университет»

Факультет прикладной математики и кибернетики

Кафедра информационных технологий

**ОТЧЕТ**

**по итогам научно-исследовательской работы**

**в 4-ом семестре**

Направление: *02.04.02 «Фундаментальная информатика и*

*информационные технологии»*

Магистерская программа: *«Информационные технологии в*

*управлении и принятии решений»*

Тема: «Сопоставление ключевых точек изображений с помощью нейронных сетей»

|  |  |
| --- | --- |
|  | **Выполнил:** студент 2 курса магистратурыИванов Артём Николаевич**Научный руководитель:**кандидат физ.-мат. наук,Сорокин Сергей Владимирович**Оценка:** |
|  Тверь – 2018 г. |  |

**ОГЛАВЛЕНИЕ**

[Введение 2](#_Toc511056394)

[Цели и задачи 2](#_Toc511056395)

[Актуальность 3](#_Toc511056396)

[Структура работы 4](#_Toc511056397)

[1. Задача сопоставления ключевых точек 5](#_Toc511056398)

[1.1. Основные определения и общее решение задачи 5](#_Toc511056399)

[1.2. Оценка состояния темы исследования 7](#_Toc511056400)

[2. Разработка архитектуры нейронной сети 8](#_Toc511056401)

[2.1. Определение и основные термины 8](#_Toc511056402)

[2.2. Общая структура сети для решения задачи 9](#_Toc511056403)

[2.3. Первоначальные вариации структур базовой нейронной сети 11](#_Toc511056404)

[2.4. Доработка структуры нейронной сети 13](#_Toc511056405)

[3. Генерация обучающей выборки 16](#_Toc511056406)

[3.1. Общий алгоритм получения набора вырезанных областей 16](#_Toc511056407)

[Заключение 18](#_Toc511056408)

[Список использованных источников 19](#_Toc511056409)

# Введение

В наше время очень много внимания уделяется совершенствованию технологий искусственного интеллекта. Большой популярностью пользуются нейронные сети, которые зачастую позволяют добиться лучших результатов[2], чем при использовании традиционных алгоритмов, как, например, в проблеме сопоставления ключевых точек изображений. Данная задача имеет широкий круг применения: создание панорам, создание стереопары и реконструкция трехмерной модели объекта по его двумерным проекциям в принципе, распознавание объектов и поиск по образцу из какой-то базы, слежение за движением объекта по нескольким снимкам, реконструкция аффинных преобразований изображений. Также следует отметить, что область знаний, рассматривающая такого рода задачи (компьютерное зрение) достаточно молода, соответственно еще нет определенного универсального метода, решающего все вышеперечисленные проблемы в полном объеме, т.е. для всех входных изображений. В тоже время существующие алгоритмы для решения данных задач постепенно совершенствуются, в том числе и с применением нейронных сетей.

## Цели и задачи

В данной научно-исследовательской работе поставлена цель:

* создать нейронную сеть для сопоставления ключевых точек изображений.

Для достижения данной цели необходимо выполнить ряд задач:

* ознакомиться с задачей сопоставления ключевых точек
* разработать архитектуру нейронной сети для сопоставления ключевых точек
* провести обучение и в случае необходимости доработку нейронной сети
* сравнить полученные результаты с известными методами
* оформить текст диссертации

В моей работе задача сопоставления ключевых точек рассматривается в контексте выделения структуры из движения (Structure from motion).

Это технология построения трехмерной структуры объекта по набору изображений (2D фотографий). Она является объектом исследования в таких областях как компьютерное зрение и зрительное восприятие (изучение зрительной системы головного мозга) [3].

## Актуальность

Данная технология построения трехмерных структур используется во многих сферах деятельности: медицине, спелеологии, геологии. Существует множество инструментов для решения данной задачи. Наиболее популярными из них являются те, которые строят трехмерные структуры по видео или фото-потоку.

Специалистам требуется стоить трехмерные модели пещер и различных предметов местности. Существующие на сегодняшний день технологии имеют ряд недостатков. Так же показывают плохой результат работы в “специфических” условиях, к которым можно отнести резкий перепад света на последовательных кадрах.

Более того, при построении топологических карт местности данная технология по сравнению с аэросъемкой позволяет добиться большей точности при минимальном количестве затрат.

Также построение трехмерной структуры используется для оценки, мониторинга, представления и реставрации памятников культурного наследия[3].

## Структура работы

Работа состоит из введения, оглавления, трёх глав, заключения и списка использованных источников.

В первой главе даётся подробное описание задачи сопоставления точек и её решения, также описываются основные термины, используемые в рассматриваемой области. Вместе с этим приводится оценка состояния темы исследования.

Во второй главе описывается решение задачи исследования посредством использования нейронных сетей. Даются основные определения, описываются используемые слои и структуры при построении и использовании нейронных сетей при решении задачи. Также в данной главе представлены основные результаты исследования.

Третья глава посвящена разработке обучающей выборки для моделей обучаемых искусственных нейронных сетей. Описывается формат наименования областей точек для формирования выборки и основные преобразования, участвующие в генерации.

# Задача сопоставления ключевых точек

Задача моей научной работы рассматривается в контексте задачи «Structure from motion» - технологии получения трехмерных моделей предметов и карт местности по данным видео и фотосъёмки. В свою очередь её можно разделить на ряд подзадач:

1. Работа детектора: поиск точек интереса (угловых);
2. Работа дескриптора: описание характеристических особенностей найденных точек
3. Поиск соответствия между дескрипторами точек для пар изображений;
4. Фильтрация ложных соответствий;
5. Определение смещения камеры между кадрами по векторам смещения пикселов;
6. Нахождение трёхмерной структуры.

Моя работа сфокусирована на 2-ой и 3-ей подзадачах, которые и выполняют сопоставление ключевых точек.

## Основные определения и общее решение задачи

Прежде чем делать дальнейшее подробное рассмотрение данных пунктов дадим определение основным терминам и понятиям, которые будут использоваться в дальнейшем:

Ключевая точка — это такая точка изображения, которую можно отличить от любой другой точки изображения. На изображении можно выделить 2 типа особых точек: углы и ребра. Самым распространенным типом таких точек являются углы. Они, в отличие от ребер, не зависят от масштаба и расположения изображения[4]. Следовательно, их можно однозначно сопоставить для пар изображений. Углы можно определить, используя различные детекторы.

Детектор — это метод получения особых точек изображения. На вход детектору подается черно-белое изображение. На выходе получается матрица, значения элементов которой определяют степень правдоподобности нахождения угла в исследуемых пикселях изображения. После выполняется отсечение участков пикселей со степенью правдоподобности, меньшей некоторого порога. Оставшиеся после отсечения точки являются особыми.

Для нахождения соответствия между точками для пар изображений к каждой угловой точке должно прилагаться её описание (дескриптор). Если дескрипторы точек на разных изображениях близки, то можно считать, что один и тот же физический объект.

Дескриптор — это небольшая окрестность точки. В идеале, дескриптор должен быть независим от масштаба и ориентации изображения. Тогда поиск соответствий сводится к обходу всех дескрипторов одного изображения и поиску наиболее близкого дескриптора на другом изображении.

Теперь опишем общий алгоритм решения задачи сопоставления точек:

Пусть нам дано: два изображения, причём они являются «похожими», т.е. получены смещением, поворотом, проекцией точки, добавлением шума, изменением положения камеры или уровня освещенности.

1. С помощью известного алгоритма компьютерного зрения находим ключевые точки двух изображений
2. Далее, используя свой оригинальный алгоритм/программный инструмент получаем дескрипторы точек изображений
3. Используя заранее определенный критерий и стандартный алгоритм (или собственный) сопоставляем дескрипторы точек изображений. Здесь можно использовать как простой перебор точек, так и метод поиска k-ближайших соседей.
4. Так как каждый дескриптор соответствует своей точке изображения, то мы получаем пары сопоставленных точек двух изображений

## Оценка состояния темы исследования

В своей работе я буду искать новый способ получения дескриптора ключевой точки изображения. При этом, полученный в результате исследования программный продукт будет проанализирован и подвержен сравнению с уже известными аналогами для решения данной задачи, ведь сейчас существует множество алгоритмов для сопоставления особых точек на изображениях. Самыми известными и широко используемыми являются SURF (Speeded Up Robust Features) и SIFT (Scale-invariant feature transform) [1]. Также надо упомянуть быструю и оптимизированную альтернативу предыдущим алгоритмам - Oriented FAST and rotated BRIEF (ORB). Он объединяет детектор локальных точек FAST и дескриптор BRIEF[6]. Некоторые из представленных выше алгоритмов являются запатентованными, и показывают нерезультативность работы в сложных условиях. Недавние исследования по нейронным сетям предлагают использовать алгоритм LIFT, который превосходит результаты своих аналогов SIFT и SURF[2]. Но при этом всегда есть возможность улучшить скорость обучения и структуру сети для достижения более точных показателей работы.

# Разработка архитектуры нейронной сети

Использование нейронных сетей для оптимизации и замены эвристических традиционных алгоритмов компьютерного зрения подтвердило свою уместность, как об этом говорилось в предыдущей главе. Поэтому, для достижения поставленной цели исследования было решено использовать искусственные нейронные сети.

## Определение и основные термины

Искусственная нейронная сеть — математическая модель, а также её программное или аппаратное воплощение, построенная по принципу организации и функционирования биологических нейронных сетей — сетей нервных клеток живого организма. Это понятие возникло при изучении процессов, протекающих в мозге, и при попытке смоделировать эти процессы. Первой такой попыткой были нейронные сети У. Маккалока и У. Питтса.

Единицей данной сети является нейрон, модель которого схематически представлена на рис.1.



Рисунок 1 Модель нейрона

Совокупность данных нейронов образует сеть из параметров весов, которые регулируются в течение обучения нейронной сети: на вход ей даются данные из выборки, которые также содержат соответствующий идеальный отклик, который должна дать сеть на эти данные. После обработки входных данных, результат сети сравнивают с желаемым откликом. На этом этапе вычисляется функция потерь (loss function), которая высчитывает ошибку. После выполняется процесс обратного распространения ошибки - распространение сигналов ошибки от выходов сети к её входам, в направлении, обратном прямому распространению сигналов в обычном режиме работы.

## Общая структура сети для решения задачи

Обращаясь к задаче исследования, нам нужно получить нейронную сеть, которая будет сравнивать два дескриптора (описания окрестности точки изображения) и вычислять, описывает ли данный дескриптор одну и ту же точку, или же разные. Предполагаемая схема поиска соответствий представлена ниже на рис.2.



Рисунок 2 Схема нахождения соответствий

Как видим, для этой задаче лучше всего подходит модель сиамской нейронной сети – которая была впервые использована для нахождения соответствия между подписями группой ученых из AT&T Laboratories в США в 1994 году [5].

Положили для начала, что модель-прототип будет выглядеть следующим образом (рис.3):



Рисунок 3 Модель-прототип

Как видно из рис.3, на вход подаются два окрестности точек изображения и параметр сигма, который указывает, принадлежат эти области одной и той же точке изображения (1) или разным (0). В основе сиамской нейронной сети лежит обычная нейронная сеть. При этом две половинки сиамской сети объединяются критерием L2-нормы, которую можно записать как:

$$\left‖x\right‖\_{2}=\sqrt{\sum\_{i}^{}x\_{i}^{2}}$$

Данная норма известна как евклидова норма, которая обычно используется для вычисления расстояния между векторами. Вычисляется как разность векторов (евклидово расстояние):

$$\left‖x\_{1}-x\_{2}\right‖\_{2}=\sqrt{\sum\_{i}^{}(x\_{1\_{i}}-x\_{2\_{i}})^{2}}$$

Для получения же дескриптора, мы программными средствами будем обрезать половинку сиамской сети, которая возвращает описательный вектор области.

## Первоначальные вариации структур базовой нейронной сети

Получив простое начальное представление модели, нам предстоит настроить её параметры и работу. Первым из таких параметров выступает половинка сиамской сети – базовая нейронная сеть.

Для начала были рассмотрены две модели для сравнения, взятые из разных источников (представлены на рис. 4).



Рисунок 4 Первоначальные структуры нейронной сети

На рис.4. первая структура а – это сверточная нейронная сеть, изначально решала близкую задачу – определяла похожие изображения (места отдыха). Была представлена программистом из Сан-Франциско Суджит Палом. Состоит из 8 слоёв:

Convolution2D – слой свёртки, выполняет преобразования схожие с оконным преобразованием фильтра: проходит по всей матрице изображения и перемножает пиксели вокруг точки на веса, затем получает суммарный отклик всех перемноженных величин. При этом веса данного фильтра меняются в процессе обучения сети.

MaxPooling2D - слой субдискретизации, он выполняет операцию максимума на тензоре (многомерной матрице), в данном случае – матрице изображения, получая максимальное значение из всех значений взятых из области, определяемой размером окна этого слоя. Например, MaxPooling2D (2x2, stride =2) имеет окно размером 2x2, соответственно берёт максимум 4-х значений рассматриваемой области 2x2.

Параметр stride - это шаг окна субдискретизации и свёртки, который позволяет обрабатывать не все пиксели изображения, а пропускать их часть, что не даёт сети создавать слишком много параметров для обучения и оптимизирует время её работы.

Вторая структура сети «b» является простой, состоит из Dense-полносвязных слоёв. Наглядно их можно представить, как переплетение выходов всех нейронов предыдущего слоя с входами нейронами последующего (рис.5).



Рисунок 5 Полносвязный слой Layer 2

Также здесь используется слои Dropout – они защищают сеть с такой переплетенной структурой от переобучения, отбрасывая часть используемых нейронов сети.

Сгенерировав обучающую выборку размером 20х20 вырезанных областей точек изображения и обучив несколько моделей сетей «а» и «b», были получены следующие результаты:

Таблица 1

|  |  |  |
| --- | --- | --- |
| **Используемая модель сети:****Название (размер обучающей выборки)** | **Тип структуры сети** | **Точность сопоставления на 500 точках первого и второго изображений** |
| choppedmySiameseArtificial\_model(239054).h5 | а | 31.800% |
| choppedmySiameseArtificial\_model\_V1(900000).h5 | а | 48.800% |
| choppedmySiameseArtificial\_model\_V2(200000).h5 | а | 34.200% |
| choppedmySiameseArtificial\_model\_V3(100000).h5 | а | 32.600% |
| choppedMySiameseNet1\_model\_V0(100000).h5 | b | 22.600% |
| choppedMySiameseNet1\_model\_V1(239054).h5 | b | 7.000% |

Как видим из таблицы 1, лучше всего справилась с задачей модель сети со структурой «а» - сверточная сеть. Но при этом точность была заметно низкой, поэтому было решено улучшить данную модель сети, изменив её параметры.

## Доработка структуры нейронной сети

Изучив статью по обучению «сверточного характеристического дескриптора ключевых точек» (Discriminative Learning of Deep Convolutional Feature Point Descriptors), разработал новую структуру сверточной сети. Данный выбор был продиктован и результатами эксперимента со структурами сетей, и ключевыми особенностями сверточных нейронных сетей[8]:

* 1. Зависимости (шаблоны), которым они обучаются инвариантны относительно переноса (translation invariant): если какая-то характерная особенность изображения переместиться в другой его угол, то сверточная сеть без труда определит её, а вот плотная (dense) сеть будет воспринимать такое изменение как новую зависимость (шаблон);
	2. Сверточные сети также могут обучиться «пространственной многоуровневости зависимостей» (spatial hierarchies of patterns): т.е. на первом слое, они могут распознавать углы, ребра, пятна, следующий слой: комбинации из рёбер и углов – многоугольники, далее – узоры, и т.д.

Новая структура сети имеет следующий вид (рис.6):



Рисунок 6 Новая структура сети

Оставляя неизменным основную последовательность слоёв, было решено поэкспериментировать с параметрами слоёв. В результате обучили порядка 7 новых моделей на основе сетей Net2-4. Итоговая сравнительная характеристика представлена на таблице 2 и соответствующими выводами по ней:

1. Наибольшая точность у моделей сети Net3;
2. Приём с переходом от 2D сверток к 1D сверткам уменьшил количество нейронов и значительно понизил точность моделей;
3. Модели, обученные на 35К и 70К различаются по точности только на 4-5%, следовательно можно предположить, что увеличение обучающей выборки в 2 раза даёт незначительный прирост точности
4. В случае с сетями Net2 и Net4 при увеличении выборки точность уменьшилась. Предполагаю, что это связано с хаотичным созданием «плохих» пар. Если данная гипотеза верная, то можно прибегнуть к «осмысленному» генерированию плохих пар для обучения.
5. При увеличении числа точек для сопоставления (например, для Net3\_model\_V1 – 1000 – 44%) точность уменьшается.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Название модели сети** | **Тип входных данных** | **Тип свёрток** | **Число нейронов** | **Размер выборки для обучения** | **Время обучения** | **Точность сопоставления\* (500 т)**  | **Точность сопоставления\* (1000 т)**  | **Точность сопоставления\* (1500 т)**  |
| choppedMySiameseNet2\_model\_V0.h5 | (50, 50, 3) цвет | 2D | 349120 | 35К | 4 ч | 25.800% | 24.100% | 22.533% |
| choppedMySiameseNet2\_model\_V1.h5 | (50, 50, 3) цвет | 2D | 349120 | 70K | 10 ч 30 м | 4.200% | 3.100% | 2.667% |
| choppedMySiameseNet3\_model\_V0.h5 | (50, 50, 1) ч/б  | 2D | 345984 | 35K | 3 ч 40 м | 45.200% | 39.200% | 36.533% |
| choppedMySiameseNet3\_model\_V1.h5 | (50, 50, 1) ч/б  | 2D | 345984 | 70K | 7 ч 40 м | 51.000% | 43.100% | 40.667% |
| choppedMySiameseNet4\_model\_V0.h5 | (50, 50) ч/б | 1D | 97568 | 35K | 10 м | 13.400% | 11.900% | 10.867% |
| choppedMySiameseNet4\_model\_V1h5 | (50, 50) ч/б | 1D | 97568 | 70K | 30 м | 10.000% | 8.000% | 8.733% |
| choppedMySiameseNet4\_model\_V2.h5 | (50, 50) ч/б | 1D | 97568 | 280K | 1 ч 36 м | 4.600% | 3.600% | 4.733% |
| **\*Точность сопоставления**  считается как ср.арифметическое из 10 полученных результатов  |  |
| Для вычисления точности используется **L2-норма и данные из набора вырезанных областей** |  |

Таблица 2.

# Генерация обучающей выборки

## Общий алгоритм получения набора вырезанных областей

Задача обученной нейронной сети - проверять на равенство дескрипторы (фрагменты изображения с ключевыми точками), принимаемые на вход. Для подготовки обучающей выборки удобно иметь уже готовый набор вырезанных областей, из которого уже посредством программы будет составляться обучающая выборка.

Для начала, мы сформируем порядок наименования областей для их дальнейшего хранения и использования (Рис.7).



Рисунок 7 Формат наименования вырезанных областей

Пары изображений на вход сети будут формироваться так:

(<1001000.jpg>, <1001001.jpg>, 1) – два изображения одной области

(<2001000.jpg>, <1001001.jpg>, 0) – два изображения разных областей.

Далее опишем алгоритм работы генератора областей:

1. Цикл по отобранным N изображениям для генерации выборки

{Ik: k = 0…N}:

1. На текущим изображении Ik находим с помощью алгоритма SIFT/SURF/ORB ключевые точки {P}
2. Вырезаем области точек определенного размера
3. Цикл по точкам j = 0…кол-во\_точек(P):
	1. В соответствии с форматом, после каждого преобразования сохраняется новая область, относящаяся к точке j изображению k и преобразованию m:

<j><k><m>.jpg

* 1. Неизмененная область
	2. Наложение шума (соль и перец)
	3. Масштабирование (фильтр Гаусса х2)
	4. Изменение яркости (х4)
	5. Поворот с шагом 5 градусов (от 0 до 360) (х71)

В итоге на одну точку изображения получаем 78 преобразований. Таким образом для одной точки можем спродуцировать 79\*78 возможных комбинаций хороших пар областей. Набор данных преобразований был выбран, чтобы сделать конечный продукт исследования эффективным в специфических природных условиях (в особенности в условиях пещер, низкой освещенности, и т.д.).

Наглядно применение фильтров к вырезанной области представлено на рис.7: 1 – исходная область, 2 – наложение шума, 3 – масштабирование, 4 – изменение яркости и 5 – преобразование поворота.



Рисунок 8 Генерация вырезанных областей

# Заключение

В этом семестре была проведена работа по созданию и настройке оптимальной модели нейронной сети для решения задачи сопоставления ключевых точек. Было спроектировано и обучено 15 различных моделей нейронных сетей для достижения поставленной цели. Также написана программа для сопоставления ключевых точек изображений и проверки точности обученных моделей. При этом был проведён анализ полученных моделей для дальнейшего их улучшения и повышения точности сопоставления.

В дальнейшем, согласно плану работ, будет произведено:

* доработка генерации обучающей выборки;
* обучение и доработка архитектуры лучшей (на текущий момент) модели нейронной сети;
* сравнение полученных результатов с известными методами;
* оформление текста диссертации.

# Список использованных источников

1. Конушин А. «Сопоставление изображений и локальные особенности» // Лекции по курсу «Введение в компьютерное зрение»,

2012, стр. 45–79.

1. Белоусов С. «LIFT: Learned Invariant Feature Transform» // Хабрахабр: интернет-сообщество индустрии высоких технологий [Электронный ресурс] <https://habrahabr.ru/post/323688/>

(дата обращения 27.12.2017)

1. Structure from motion // Wikipedia, The Free Encyclopedia [Электронный ресурс] <https://en.wikipedia.org/w/index.php?title=Structure_from_motion&oldid=816884383> (дата обращения 27.12.2017)
2. Обзор дескрипторов ключевых точек // НОУ «ИНТУИТ» [Электронный ресурс] [http://www.intuit.ru/studies/courses/10621/1105/lecture/17983](http://www.intuit.ru/studies/courses/10621/1105/lecture/17983?page=2)

(дата обращения 9.04.2018)

1. Bromley J., Guyon I., LeCun Y., Sickinger E., Shah R. «Signature Verification using a "Siamese" Time Delay Neural Network», 1994
2. ORB (Oriented FAST and Rotated BRIEF) // OpenCV-Python Tutorials [Электронный ресурс] <https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_feature2d/py_orb/py_orb.html#orb>

(дата обращения 9.04.2018)

1. Simo-Serra E., Trulls E., Ferraz L., Kokkinos I., Fua P., Moreno-Noguer F. Discriminative Learning of Deep Convolutional Feature Point Descriptors // ICCV (2015)

<http://hi.cs.waseda.ac.jp/~esimo/publications/SimoSerraICCV2015.pdf>

1. Chollet F. Deep Learning with Python. NY, 2018.