ISSN 2079-3316 ПРОГРАММНЫЕ СИСТЕМЫ: ТЕОРИЯ И ПРИЛОЖЕНИЯ №4(35), 2017, с. 327–345

УДК 004.93:004.89

А. Ю. Буйко, А. Н. Виноградов

Выявление действий на видео с помощью рекуррентных нейронных сетей

Аннотация. В настоящей работе рассмотрено применение методов компьютерного зрения и рекуррентных нейронных сетей для решения задачи выявления и классификации действий на видео.

В статье приводится описание подхода, применённого авторами для анализа видеофайлов. Рекуррентные нейронные сети выступают в качестве классификатора. На вход классификатору передаются мешки слов, которые являются гистограммами низкоуровневых действий. Гистограммы представляют собой наборы дескрипторов кадров видеофайлов. Для поиска дескрипторов на изображениях используются алгоритмы SIFT, ORB, BRISK, AKAZE.

Ключевые слова и фразы: компьютерное зрение, дескрипторы, мешки слов глубинное обучение, рекуррентные нейронные сети, сети долгой краткосрочной памяти, анализ видео.

Введение

Стремительное развитие современных компьютерных технологий привело к появлению мощных и доступных вычислительных устройств, позволяющих реализовывать и выполнять сложные алгоритмы, требующие значительных вычислительных мощностей. Благодаря этому стала возможна практическая реализация различных методов искусственного интеллекта, в том числе – эффективных методов машинного обучения, позволяющих заменить человека в различных сферах деятельности, требующих длительной монотонной обработки информации. Обратной стороной технического прогресса, в этом направлении, является стремительно возрастающий объём накапливаемой информации, из которой анализируется менее одного процента.

Работа выполнена в рамках НИР «Исследование и разработка методов машинного обучения для обнаружения аномалий», номер гос. регистрации 0077-2016-0002.

DOI: 10.25209/2079-3316-2017-8-4-327-345

Одним из самых быстрорастущих сегментов данных является видеоинформация. По данным компании CISCO [1] в течение ближайших четырёх лет потребление видео (включая IP VOD – видео по запросу) займёт 82% от всего интернет-трафика, что составит 228 Экзабайт данных (для сравнения – в 2016 году объём видеотрафика составлял 73% и 70 Экзабайт соответственно). По данным исследований, проведенных той же компанией CISCO в 2014 году – к 2019 году потребуется более 5 миллионов лет, чтобы просмотреть в режиме реального времени весь объем видеофайлов с IP-камер, отснятых за месяц. Таким образом, всё более актуальной становится задача автоматического анализа видеопотоков и выделения из них значащей семантической информации.

Например, видео с дорожных и уличных камер могут содержать как моменты рядовых событий, так и моменты правонарушений, которые требуют своевременной фиксации и передачи в правоохранительные органы. Автоматическая идентификация, при помощи камер наружного наблюдения, преступников и террористов по характерным жестам, позволит существенно упростить работу правоохранительных органов и, потенциально, спасти множество жизней. Ещё одной важной задачей является анализ видео со спортивных мероприятий, что позволит сделать судейскую оценку наиболее объективной, исключив человеческий фактор. Всё это требует распознавания действий на видео.

На сегодняшний день, не существует готового универсального решения таких задач, несмотря на то, что многие крупные компании IT индустрии, в том числе Google, Facebook, NVidia и др., активно ведут фундаментальные и прикладные исследования в области машинного обучения.

В рамках данной работы был проведён анализ существующих методов, обработки и классификации видеопотоков и выявления на них определённых типов действий. Видеопоток представляет собой последовательность изображений, сопровождаемых аудиодорожкой, однако в этом исследовании работа велась только с визуальными данными, аудиопотоки не анализировались.

По результатам исследований был предложен подход, использующий в качестве классификатора рекуррентную нейронную сеть (РНС, англ. Recurrent neural network; RNN) с долгой краткосрочной памятью (англ. Long short-term memory; LSTM) – LSTM РНС, одним из преимуществ которой является способность обработки длинных последовательностей информации.

В качестве входных данных сети были использованы так называемые «мешки слов» (Bag of Words), которые условно представляют собой матрицы частотных признаков, что позволило уменьшить объём обрабатываемых данных. Признаки представлены дескрипторами

кадров из видео. В данной работе рассматривались следующие алгоритмы, вычисляющие дескрипторы: SIFT, BRISK, ORB, AKAZE.

1. Существующие подходы к решению задачи выявления движений на видео

В большинстве работ, связанных с распознаванием действий на видео, принято выделять два подхода: выявление низкоуровневой и высокоуровневой информации о движениях.

При низкоуровневом подходе система-анализатор имеет представление о некотором наборе базовых элементов, либо движений (low-level labels), из которых можно составить различные сложные действия. Например, движение ног может свидетельствовать о ходьбе, беге игре в футбол и т.д. Потенциально, низкоуровневый подход сможет предоставлять более точную семантическую информацию используя ограниченный набор базовых движений. К недостаткам такого подхода относят сложность реализации, необходимость составления сложных взаимосвязей и структур для эффективной работы подхода и необходимость весьма значительных вычислительных мощностей.

Примерами работ с таким подходом являются [2] и [3]. Обе работы были протестированы на футбольных видео с 4 классами действий. В работе [4] удалось получить точность классификации на уровне 52,75% с помощью k-NN классификатора и 73,25% при использовании системы опорных векторов (SVM-based). В работе [5] были получены схожие результаты, однако, в качестве классификатора использовались PHC, что позволило улучшить точность классификации до 74%.

Высокоуровневый подход предполагает моделирование переходов между кадрами, без анализа базовых действий. Другими словами, система-анализатор работает с целостным образом. К преимуществам такого метода можно отнести относительную простоту реализации и требующиеся меньшие вычислительные мощности по сравнению с низкоуровневым подходом. Главным недостатком подхода является ограничения памяти, то есть системе требуется запомнить значительный набор примеров высокоуровневых действий для эффективного распознавания. Из-за бессчётного множества существующих действий, даже потенциально, не получится создать универсальную систему с высокоуровневым подходом. В работах [4–9] использовался высокоуровневый подход. В частности, в [7] использовалась LSTM RNN для структурирования теннисных видеороликов с точностью классификации 71.1%. В [5, 8, 9] использовались свёрточные нейронные сети (англ. convolutional neural network, CNN). CNN выступали в качестве преобразователя изображений во входные данные, т.е. условно являются первым слоем нейронной сети, которая работает с видеофайлами. Наилучший показатель точности был достигнут в работе [10] на уровне 89.1%.

В данной работе использовался низкоуровневый подход, схожий с работами [4, 5]. В этих работах видео были представлены наборами SIFT-дескрипторов. Для увеличения точности, в качестве классификатора была использована LSTM РНС. Также было проверено влияние используемых входных данных на точность классификации и примере различных дескрипторов.

2. Выбор методов

Для успешного решения задачи распознавания действий на видео необходимо выбрать наиболее удобный способ представления входных/выходных данных и подобрать оптимальный классификатор, позволяющий достичь наилучшей точности предсказаний.

2.1. Метод представления данных: Bag of Words

В данной работе в качестве метода представления данных был выбран Мешок слов (Bag of Words, BoW)[11]. Несмотря на то, что этот метод первоначально был создан для представления текстовых данных, он также был успешно использован в работах [7, 8, 12] для представления визуальных данных. Применительно к задаче классификации изображений в некоторых работах [7, 8] используется отдельный термин «мешок визуальных-слов» (Bag of Visual-Words), но большинство исследователей используют общее наименование BoW. Мешок слов, также иногда называемый векторной моделью (Vector space model) [13] представляет собой популярный подход, который формирует из набора локальных дескрипторов изображений некоторое представление данных, которое может быть использовано для поиска и классификации. На сегодняшний день существует целый ряд его вариантов, используемых в различных задачах представления изображений, например:

- Kernel Codebook encoding (KCB) [14],
- Locality Constrained Linear Coding (LLC) [15],
- Improved Fisher encoding (IFE) [16],

В работах [17, 18] приведено сравнение вышеперечисленных методов. Выбор конкретного метода представления данных при проведении экспериментальных исследований был обусловлен исключительно удобством реализации, однако в дальнейшем планируется исследовать его влияние на точность классификации.

Основная идея BoW заключается в том, чтобы представлять изображения с помощью гистограммы встречаемости визуальных слов, соответствующих каждому набору локальных признаков, извлеченных из изображений. Локальные признаки представлены дескрипторами.

Дескриптором называется числовой или бинарный вектор описывающий ключевую точку, который отражает особенности её окрестности, какие как форма, цвет или текстура. Ключевой точкой, в свою очередь, называют некоторую точку изображения, которая уникально характеризует ближайшую окрестность изображения.

Как уже было упомянуто, в данной работе рассматривались дескрипторы, полученные с помощью алгоритмов: SIFT, BRISK, ORB, AKAZE.

Алгоритм SIFT строит пирамиды разности гауссиан, находит экстремумы (самые яркие и тёмные точки), фильтрует их и составляет массив ключевых точек. Далее, вычисляется SIFT- дескриптор, который является вектором с размерностью 128 значений, где каждое значение является числом типа float.

Алгоритмы ORB и BRISK одинаковым образом находят ключевые точки (с помощью метода FAST (Features from Accelerated Segment Test)), однако по-разному вычисляют дескрипторы. ORB-дескрипторы — это бинарные векторы с размерностью 32, а BRISK-дескрипторы — бинарные векторы с размерностью 64. Очевидно, что первые вычисляются значительно быстрее, но содержат меньше информации в сравнении со вторыми, что часто влияет на точность распознавания.

Алгоритм AKAZE был предложен позже вышеперечисленных (в 2013). Для поиска ключевых точек, алгоритм применяет методы алгоритма FED-Fast Explicit Diffusion на пирамидальной схеме, что позволяет построить нелинейную многомасштабную пирамиду. Применение нелинейного коэффициента масштабирования позволяет увеличить скорость нахождения нужной ключевой точки по сравнению с Гауссовой пирамидой. AKAZE-дескрипторы вычисляются схожим с BRISK образом, однако благодаря некоторым изменениям вычисляются быстрее.

2.2. Классификатор: LSTM RNN

Искусственная нейронная сеть — это математическая модель сети биологических нейронов, состоящая из входного слоя, скрытого слоя/слоёв и выходного слоя. Каждый слой состоит из одного или нескольких нейронов, которые состоят из наборов синапсов, сумматоров, функций активации и пороговых элементов.

Одной из разновидностей сетей является рекуррентная нейронная сеть (РНС, англ. Recurrent neural network; RNN). РНС отличаются от традиционных искусственных нейронных сетей наличием обратных связей (как внутри, так и между слоями), что позволяет при анализе текущего набора входных значений учитывать результаты предыдущих итераций. Данная особенность позволяет говорить о наличии

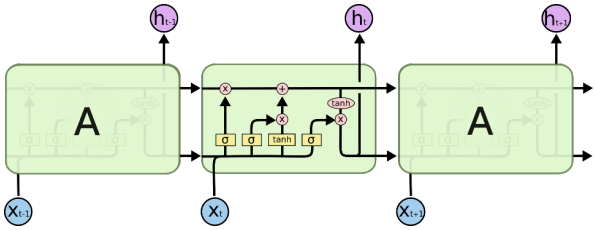

Рис. 1. Схема архитектуры рекуррентной сети с LSTM-ячейкой[19]

«эффекта памяти», позволяющего анализировать последовательности подаваемых на вход данных. Такая сеть имеет существенный недостаток, при работе с длинными последовательностями «эффект памяти» ограничен и, порой, важные данные теряются.

Рекуррентная нейронная сеть с «долгой краткосрочной памятью» (англ. Long short-term memory; LSTM) — это специфическая модификация классической РНС, которая способна учитывать долгосрочные зависимости между подаваемыми на вход данными (в обычных рекуррентных нейронных сетях при увеличении расстояния между двумя подаваемыми на вход последовательностями данных зависимость ослабевает), это достигается за счёт того, что LSTM ячейка не использует функцию активации внутри своих рекуррентных слоёв. Это приводит к тому, что важные значения не размывается во времени при использовании метода обратного распространения ошибки (англ. Backpropagation) при обучении сети. Важная информация поступает в запоминающую ячейку и удаляется оттуда в соответствии с заранее заданными правилами.

На рис. 1 изображена архитектура рекуррентной сети с LSTM-ячейкой.

3. Постановка задачи

Предполагается, что изначально имеется ограниченный набор известных действий (классов) и размеченный набор видеофайлов, где каждое видео содержит фон и объект, совершающий некоторое действие, например, прогулка, бросок мяча и т.д. Постановка задачи заключается в том, чтобы научиться классифицировать видео по критерию действия, совершаемого объектом, а именно, найти наиболее вероятный класс действия на видео.

На рис. 2 представлена общая схема предложенного подхода.

В данной работе был выбран низкоуровневый подход, т.к. он лучше, чем высокоуровневый подходит для решения поставленной задачи,

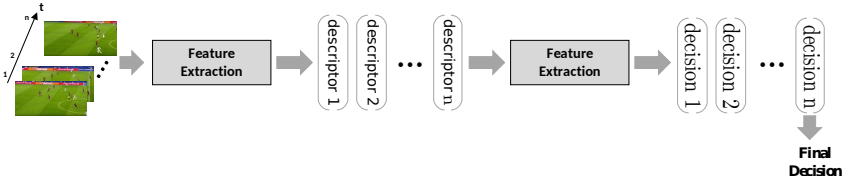

Рис. 2. Предлагаемая схема классификатора

при условии ограниченного числа известных классов. Низкоуровневый подход требует значительных вычислительных ресурсов только для вычисления базовых элементов, однако, в дальнейшем, при распознании, требуется производить меньше операций по сравнении с высокоуровневым подходом.

Итак, для начала необходимо составить некоторый набор базовых элементов.

Сначала, чтобы уменьшить объём обрабатываемой информации, все видео переводились в последовательность векторов с помощью алгоритмов поиска дескрипторов. Такие алгоритмы ошибаются при поиске ключевых точек на изображениях фона, где нет явных границ цвета, или фрактальных изображениях. Однако, в нашем случае, все видео с действиями должны содержать отчетливо выделяемые объекты на фоне, т.е. на кадрах присутствуют границы. Таким образом, дескрипторы позволили сохранить основную информацию с минимальными потерями. В результате, один видеофайл будет представлять собой массив из дескрипторов кадров f_n, где f_i — i-ый кадр видео. Для того чтобы уменьшить количество дескрипторов, что необходимо для уменьшения количества вычислительных операций, использовался алгоритм RANSAC, который отсеивал точки, которые не относятся к действующему объекту.

Далее, для каждого кадра составляется «справочник», состоящий из близлежащих дескрипторов. Затем, для каждого видео связываются одинаковые значения из «справочников» изображений одного видео и составляется «мешок слов», который представляет собой упорядоченный массив из наиболее часто встречающихся дескрипторов (визуальный контент) и массив смещений (набор переходов между кадрами). Полученные «мешки слов» являются набором базовых движений.

Теперь необходим классификатор. Рекуррентная нейронная сеть с LSTM-ячейкой способна работать с длинными последовательностями данных, поэтому был выбран этот классификатор. Далее, такая нейронная сеть обучалась методом обратного распространения ошибки по времени. На вход сети подавались видео из обучающей выборки

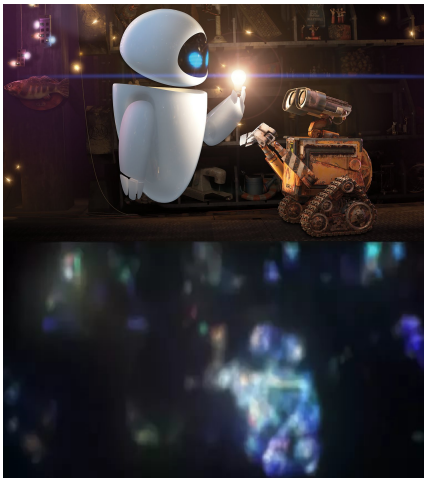

Рис. 3. Кадр из анимационного фильма ВАЛЛ-И (наверху) с визуализацией дескрипторов (снизу)

представленные «мешками слов и на выходе были известны классы, к которым принадлежали эти видео. После некоторого числа итераций (подбирается эмпирически), обученный классификатор, которому на вход подаётся видео (в формате «мешка слов»), должен выдать вероятностную принадлежность тому или иному известному классу.

Итоговая схема классификатора для задачи анализа движения на видео состоит из следующих этапов:

1. Векторизация видеофайла (вычисление дескрипторов их фильтрация).
2. Вычисление сдвигов и составление мешка слов.
3. Подача дескрипторов на входной слой.
4. Работа рекуррентной нейронной сети:
   (*a*) работа слоя LSTM (важный элемент сети, выполняющий роль памяти);
   (*b*) работа слоя Dropout (необходим для предотвращения переобучения);
   (*c*) работа слоя pooling (технический усредняющий слой);
   (*d*) работа Softmax слоя (функция, выполняющая роль классификатора).
5. Выходной слой, выведение результата.

Для наглядности на рис. 3 показан исходный кадр (наверху) с визуализацией дескрипторов (снизу).

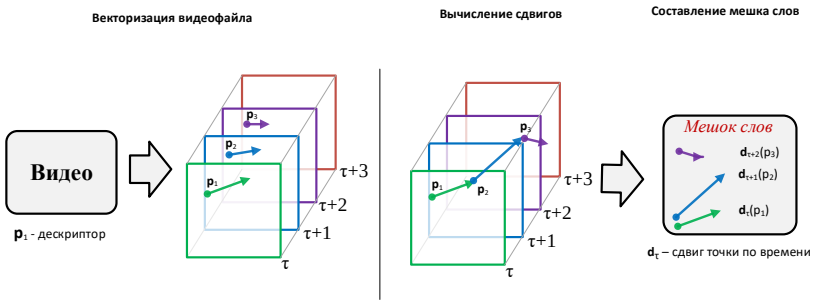

Рис. 4. Схема векторизации видеофайла, вычисления сдвигов и составления мешка слов

На рис. 4 отображены пункты 1 и 2 итоговой схемы классификатора.

4. Практическая реализация

4.1. Набор данных

После поиска доступных наборов видеофайлов было принято решение произвести тестирование схемы на самом распространённом наборе UCF-11, который состоит из сотен видео, принадлежащих 11 классам. Также доступна расширенная версия: UCF-101 [20], включающая 101 класс видеофайлов. На первом наборе была проведена отладка работы алгоритма и сравнение точности распознавания разных схем, а на втором было проведено тестирование на эффективность работы лучшей схемы, полученной на первом наборе.

Набор видео для обучения из UCF-11 включает 1168 видео файлов, а UCF-101 13480 видео файлов. Тестирование проводилось на предложенных тестирующих выборках.

Именно эти наборы встречались в большинстве работ в исследуемой области, поэтому было проще сравнивать эффективность работы. В некоторых работах, также, встречались наборы, 4-socceraction [2–5] HMDB-51 [7] и собственные подборки.

4.2. Распределённые вычисления

Проведение экспериментальных исследований для проверки предложенных в работе подходов и сравнения их с уже имеющимися,

Таблица 1. Результаты, полученные на наборе UCF-11

Предлагаемый метод	Средняя точность предсказаний
BoW(SIFT) + k-NN	22,75 %
BoW(SIFT) + SVM	33,25 %
BoW(ORB) + LSTM-PHC	36,00 %
BoW(BRISK) + LSTM-PHC	36,25 %
BoW(SIFT) + LSTM-PHC	43,00 %
BoW(AKAZE) + LSTM-PHC	**47,25** %

потребовало использования больших вычислительных мощностей, ввиду большого количества обрабатываемых данных (извлекаемых дескрипторов) и высокой вычислительной сложности алгоритмов обучения РНС. Ресурсов одной ПЭВМ оказалось недостаточно для проведения экспериментов за приемлемое время, поэтому было принято решение объединить несколько вычислительных узлов в одну сеть. Распределённые вычисления были реализованы с помощью средств из библиотеки Tensorflow. Для обоснования выбора средства распределённых вычислений был проведён эксперимент на наборе данных MNIST. Это набор включал 50000 изображений рукописных арабских цифр для обучения и 10000 для проверки точности распознавания. В качестве классификатора выступала логистическая регрессия. Тестирование проводилось с использованием Tensorfow и вычислением на GPU; и стандартной библиотеки Scikit-learn и вычислением на CPU. По времени выполнения Tensorfow превзошёл Scikit-learn в 7,36 раз.

5. Полученные результаты

Изначально, для изучения эффективности классификаторов были проверены и сравнены уже предложенные схемы из работы [2]. Затем, было проведено сравнение между одним классификатором (LSTM РНС) и различными дескрипторами.

В таблице 1 отображены результаты точности работы различных комбинаций классификаторов и дескрипторов. Видно, что схема с использованием нейронной сети в качестве классификатора в значительной степени превосходит подход, основанный на k-NN, и дает лучшие результаты, чем на основе SVM (на 20.25% и 9.75% соответственно).

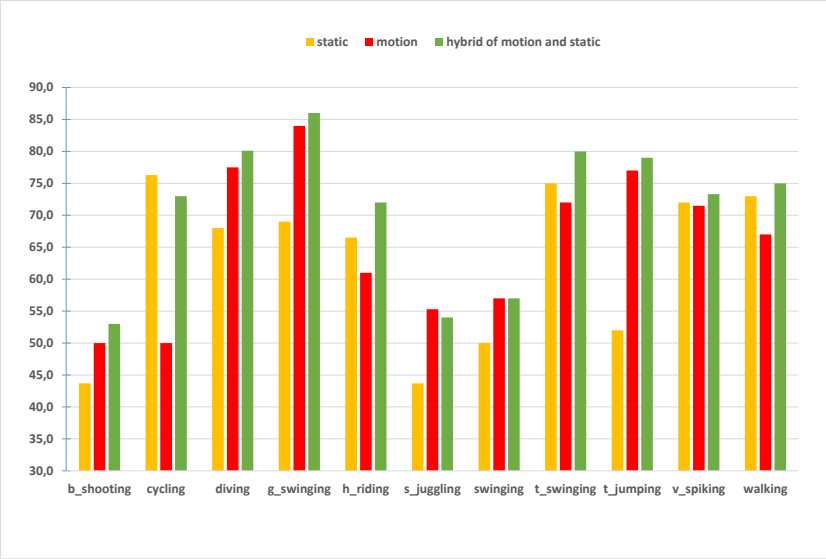

Рис. 5. Сравнение точности распознавания по классам на расширенной UCF-11

Затем LSTM-PHC была протестирована в комбинациях с дескрипторами SIFT, ORB, BRISK и AKAZE. При использовании дескрипторов ORB и BRISK был достигнут почти одинаковый результат, с незначительной разницей в 0.25%. Использование дескрипторов SIFT позволило достичь 43% точности.

Сеть с использованием дескрипторов AKAZE смогла достигнуть точности на уровне 47,25%, что на 4,25% превосходит результаты, соответствующие использованию дескрипторов SIFT. Затем, выборка UCF-11 была расширена; в среднем было добавлено по 10 видео к каждому классу.

Далее для сравнения будет использоваться результат, полученный с помощью схемы с дескрипторами AKAZE, так как она показала лучшую точность распознавания среди всех проверенных комбинациях.

На рис. 5 представлено сравнение полученных результатов точности для каждого класса из UCF-11 по типам (в среднем добавлено по 10 видео к каждому классу). В результате получилось, что средняя точность распознавания на расширенной UCF-11 для статичных, движущихся и гибридных видео соответствует: 65.3%, 63.0% и 71.2%.

	b_sh	cy	div	g_sw	h_rid	s_jug	sw	t_sw	t_ju	v_sp	wa
b_shooting	53,0	4,0	1,0	8,0	2,0	0,0	3,0	17,0	0,0	6,0	6,0
cycling	5,0	73,0	3,0	3,0	11,0	0,0	2,0	0,0	2,0	0,0	1,0
diving	4,0	3,0	81,0	0,0	0,0	1,0	2,0	0,0	0,0	6,0	3,0
g_swinging	7,0	0,0	0,0	86,0	0,0	1,0	0,0	5,0	0,0	1,0	0,0
h_riding	1,0	13,0	0,0	0,0	72,0	1,0	2,0	2,0	2,0	6,0	1,0
s_juggling	7,0	11,0	1,0	4,0	1,0	54,0	5,0	9,0	7,0	1,0	0,0
swinging	4,0	12,0	2,0	1,0	2,0	0,0	57,0	1,0	13,0	8,0	0,0
t_swinging	6,7	1,3	1,3	3,3	0,7	0,0	0,7	80,0	0,0	3,3	1,3
t_jumping	1,0	0,0	0,0	0,0	1,0	6,0	10,0	0,0	79,0	1,0	0,0
v_spiking	9,9	1,0	1,0	0,0	0,0	1,0	0,0	7,9	0,0	73,3	5,9
walking	6,0	2,0	2,0	1,0	0,0	1,0	0,0	7,0	0,0	6,0	75,0

Рис. 6. Таблица предсказаний предлагаемой схемы на расширенной UCF-11 для смешанных действий

На рис. 6 изображена таблица, где представлены результаты предсказаний для расширенной UCF-11 с гибридными видео. Из этой таблицы видно, что наибольшей точности удалось в распознавании игры в гольф (g_swinging) на уровне 86%. Хуже всего распознавался бросок баскетбольного мяча (b_shooting): 54%. Классификатор в 17% случаях путал игру в баскетбол с игрой в теннис (t_ sw).

Точность распознания, зависела от используемых для обучения видео. Так, например, видео с бросками мяча сильно различались, как по технике броска, так и по заднему фону (уличная площадка, помещение), в то время, как игра в гольф на всех видео проходила на лужайке.

В ходе серии тестов выяснилось, что точность классификации нейронных сетей сильно зависит от количества примеров, на которых они были обучены. Поэтому было принято решение сравнить точность классификации по количествам примеров, на которых обучалась используемая LSTM-PHC.

Корреляция точности классификации по количеству видео в классе, видна на рис. 7.

Рис. 7. График зависимости точности распознавания действий от числа используемых для обучения видео в среднем на класс (UCF-11 расширенная)

5.1. Сравнение с аналогами

Далее полученный результат (вариант с дескрипторами AKAZE) сравнивался с несколькими другими известными схемами, которые называют себя “state-of-art”: [7–9, 21]. Результаты приведены в таблице 2.

Предлагаемый метод работает на 4,3% точнее, чем модель LRCN, которая также использовала комбинацию LSTM PHC со свёрточными входными слоями. Кроме того, текущий метод работает лучше, чем функции C3D, использующие свертку 3D. Однако, в случае, когда к функции C3D добавляется fc6 (полно-связанный слой, предложенный в [21]), метод работает точнее, чем использованный в текущей работе. Очевидно, это связано с тем, что такой слой работает с дескрипторами кадров с высокой размерностью (4096), которые позволяют использовать для анализа большее количество информации. Однако, использование такого слоя негативно сказывается на затрачиваемых ресурсах при вычислении.

Заключение

Предложенная схема показала свою работоспособность и, в среднем, смогла распознать 3 видео из 4-х.

Таблица 2. Сравнение со “state-of-art” решениями, согласно [7]

Method	UCF-101
Spatial Convolutional Net [8]	73.0
C3D [21]	72.3
C3D + fc6 [21]	76.4
LRCN [22]	71.1
Composite LSTM Model [7]	75.8
Предлагаемый метод (LSTM с AKAZE)	75.3
Temporal Convolutional Net [8]	83.7
Two-stream Convolutional Net [9]	88.0
Multi-skip feature stacking [10]	**89.1**

К достоинствам предложенного подхода можно отнести меньшее время работы по сравнению с методами, использующими свёрточные нейронные сети. Это достигается за счёт того, что для вычисления представления видео требуется меньшее количество операций. Среди имеющихся недостатков можно выделить: 1) сильную зависимость от количества классов действий (чем больше классов известно, тем больше информации надо хранить и операций выполнить), 2) высокие требования к вычислительным ресурсам, несмотря на использование различных методов оптимизации, 3) потеря части информации из-за использования дескрипторов.

В дальнейшем планируется провести исследование других существующих методов и подходов, применимых к исследуемой задаче включая различные методы представления данных. Судя по работе [8], нейронные сети, обучающиеся без учителя, например, Autoencoder, показали достаточно высокую точность. Поэтому предполагается исследовать применимость недавно появившегося алгоритма Улучшенной Самоорганизующейся Растущей Нейронной Сети (ESOINN). На текущий момент удалось найти небольшое число публикаций по самой сети и пока ни одной, описывающей работу такой сети с видео. Несмотря на это, данная нейронная сеть демонстрирует хорошую работу в задачах разметки данных и кластеризации, что позволяет предположить, что её использование в задаче классификации действий на видео позволит улучшить уже достигнутые результаты.

Исходя из того, что методы глубокого обучения сильно нуждаются в вычислительных ресурсах с одной стороны, и являются хорошо распараллеливаемой задачей, с другой, предполагается также провести

анализ существующих альтернативных (не Фон-Неймановских) вычислительных архитектур, позволяющих более эффективно использовать преимущества параллелизма.

Список литературы

[1] *VNI Global Fixed and Mobile Internet Traffic Forecasts*, URL: https://www.cisco.com/c/en/us/solutions/service-provider/visual-networking-index-vni/index.html ↑ 328

[2] A. Ekin, A. Tekalp, R. Mehrotra. “Automatic Soccer Video Analysis and Summarization”, *IEEE Transactions on Image Processing*, **12**:7 (2003), pp. 796–807. ↑ 329,335,336

[3] Y. Gong, T. Lim, H. Chua. “Automatic Parsing of TV Soccer Programs”, IEEE International Conference on Multimedia Computing and Systems, 1995, pp. 167–174. ↑ 329,335

[4] L. Ballan, M. Bertini, A. Del Bimbo, G. Serra. “Action categorization in soccervideos using string kernels”, CBMI ’09. Seventh International Workshop on ontent-Based Multimedia Indexing (3–5 June 2009, Chania, Crete). ↑ 329,330,335

[5] M. Baccouche, F. Mamalet. *Action Classification in Soccer Videos with Long neural networks* Technical Report IDSIA-07-02, IDSIA/USI-SUPSI. ↑ 329,330,335

[6] X. Glorot, Y. Bengio. “Understanding the difficulty of training deep feedforward neural networks”, AISTATS 2010 (13–15 May 2010, Chia Laguna Resort, Sardinia, Italy), *Proceedings of Machine Learning Research*, **9**, pp. 249–256. ↑ 329

[7] N. Srivastava, E. Mansimov, R. Salakhutdinov. *Unsupervised Learning of Video Representations using LSTMs*, 2015, arXiv: 1502.04681. ↑ 329,330,335,339,340

[8] K. Simonyan, A. Zisserman. “Two-stream convolutional networks for action recognition in videos”, NIPS 2014 (8-13 December 2014, Palais des Congrès de Montréal, Montréal, Canada), *Advances in Neural Information Processing Systems*, **27**. ↑ 329,330,339,340

[9] K. Simonyan, A. Zisserman. *Very deep convolutional networks for large-scale image recognition*, 2014, arXiv: 1409.1556. ↑ 329,339,340

[10] Zh.-Zh. Lan, M. Lin, X. Li, A. G. Hauptmann, B. Raj. *Beyond gaussian pyramid: Multi-skip feature stacking for action recognition*, 2014, arXiv: 1411.6660. ↑ 329,340

[11] P. Koniusz, Fei Yan, Ph.-H. Gosselin, K. Mikolajczyk. «Higher-Order Occurrence Pooling for Bags-of-Words: Visual Concept Detection», *IEEE Transactions on Pattern Analysis & Machine Intelligence*, **39** (2017), с. 313–326. ↑ 330

[12] G. J. Brostow, J. Shotton, J. Fauqueur, R. Cipolla. «Segmentation and recognition using structure from motion point clouds», Computer Vision – ECCV 2008, Lecture Notes in Computer Science, т. **5302**, Springer, Berlin–Heidelberg, 2008, с. 44–57. ↑ 330

[13] V. Ramanathan, Sh. Mishra, P. Mitra. «Quadtree decomposition based extended vector space model for image retrieval», 2011 IEEE Workshop on Applications of Computer Vision (WACV) (5–7 Jan. 2011, Kona, HI, USA), с. 139–144. ↑ 330

[14] J. C. van Gemert, J. M. Geusebroek, C. J. Veenman, A. W. M. Smeulders. "Kernel codebooks for scene categorization", Computer Vision – ECCV 2008, Lecture Notes in Computer Science, vol. **5304**, Springer, Berlin–Heidelberg, pp. 696–709. ↑ 330

[15] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong. "Locality-constrained linear coding for image classification", 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (13–18 June 2010, San Francisco, CA, USA), 9 p. ↑ 330

[16] J. Sánchez, F. Perronnin, T. Mensink, J. Verbeek.. "Image Classification with the Fisher Vector: Theory and Practice", *International Journal of Computer Vision*, **105**:3, pp. 222–245. ↑ 330

[17] K. Chatfield, K. Simonyan, A. Vedaldi, A. Zisserman. "Return of the devil in the details: Delving deep into convolutional nets", British Machine Vision Conference BMVC 2014 (1–5 September, 2014, Nottingham, UK), URL: http://www.bmva.org/bmvc/2014/files/paper054.pdf ↑ 330

[18] G. Csurka, F. Perronnin. "Fisher Vectors: Beyond Bag-of-Visual-Words Image Representations", VISIGRAPP 2010: Computer Vision, Imaging and Computer Graphics. Theory and Applications, Communications in Computer and Information Science, vol. **229**, Springer, Berlin–Heidelberg, pp. 28–42. ↑ 330

[19] Ch. Olah. *Understanding LSTM Networks*, August 27, 2015, URL: http://colah.github.io/posts/2015-08-Understanding-LSTMs/ ↑ 332

[20] Kh. Soomro, A. R. Zamir M. Shah. *UCF101: A Dataset of 101 Human Action Classes From Videos in The Wild*, CRCV-TR-12-01, 2012. ↑ 335

[21] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, M. Paluri. *C3D: generic features for video analysis*, 2014, arXiv: 1412.0767. ↑ 339,340

[22] J. Donahue, L.-A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, T. Darrell. *Long-term recurrent convolutional networks for visual recognition and description*, 2014, arXiv: 1411.4389. ↑ 340

Рекомендовал к публикации *к.т.н. Е. П. Курпиев*

Пример ссылки на эту публикацию:

А. Ю. Буйко, А. Н. Виноградов. «Выявление действий на видео с помощью рекуррентных нейронных сетей», *Программные системы: теория и приложения*, 2017, **8**:4(35), с. 327–345.

URL: http://psta.psiras.ru/read/psta2017_4_327-345.pdf

Об авторах:

Александр Юрьевич Буйко

Студент 1 курса магистратуры Российского университета дружбы народов, Факультет физико-математических и естественных наук, кафедра информационных технологий. Область научных интересов: распознавание образов, искусственные нейронные сети, машинное обучение, вычисления на графических процессорах

e-mail: sas6092@yandex.ru

Андрей Николаевич Виноградов

к.ф.-м.н., Заместитель руководителя Исследовательского центра искусственного интеллекта ИПС им. А. К. Айламазяна РАН. Доцент кафедры информационных технологий Российского университета дружбы народов. Область научных интересов: искусственный интеллект и принятие решений, интеллектуальный анализ данных и распознавание образов

e-mail: andrew@andrew.botik.ru

Aleksandr Buyko, Andrey Vinogradov. *Action recognition on video using recurrent neural networks.*

Abstract. In this paper, we consider the application of computer vision and recurrent neural networks to solve the problem of identifying and classifying actions on video. The article describes the approach taken by the authors to analyze video files. Recurrent neural networks uses as a classifier. The classifier takes data in a "bags of words" format that describes low-level actions. The histograms contained in a "bags of words" are represented by sets of video file descriptors. Next algorithms are used to search for descriptors: SIFT, ORB, BRISK, AKAZE. (*In Russian*).

Key words and phrases: computer vision, descriptors, bags of words, deep learning, recurrent neural networks, long short-term memory networks, video analysis.

References

[1] *VNI Global Fixed and Mobile Internet Traffic Forecasts*, URL: https://www.cisco.com/c/en/us/solutions/service-provider/visual-networking-index-vni/index.html

[2] A. Ekin, A. Tekalp, R. Mehrotra. "Automatic Soccer Video Analysis and Summarization", *IEEE Transactions on Image Processing*, **12**:7 (2003), pp. 796–807.

[3] Y. Gong, T. Lim, H. Chua. "Automatic Parsing of TV Soccer Programs", IEEE International Conference on Multimedia Computing and Systems, 1995, pp. 167–174.

[4] L. Ballan, M. Bertini, A. Del Bimbo, G. Serra. "Action categorization in soccervideos using string kernels", CBMI '09. Seventh International Workshop on ontent-Based Multimedia Indexing (3–5 June 2009, Chania, Crete).

[5] M. Baccouche, F. Mamalet. *Action Classification in Soccer Videos with Long neural networks* Technical Report IDSIA-07-02, IDSIA/USI-SUPSI.

[6] X. Glorot, Y. Bengio. "Understanding the difficulty of training deep feedforward neural networks", AISTATS 2010 (13–15 May 2010, Chia Laguna Resort, Sardinia, Italy), *Proceedings of Machine Learning Research*, **9**, pp. 249–256.

[7] N. Srivastava, E. Mansimov, R. Salakhutdinov. *Unsupervised Learning of Video Representations using LSTMs*, 2015, arXiv: 1502.04681.

[8] K. Simonyan, A. Zisserman. "Two-stream convolutional networks for action recognition in videos", NIPS 2014 (8-13 December 2014, Palais des Congrès de Montréal, Montréal, Canada), *Advances in Neural Information Processing Systems*, **27**.

[9] K. Simonyan, A. Zisserman. *Very deep convolutional networks for large-scale image recognition*, 2014, arXiv: 1409.1556.

[10] Zh.-Zh. Lan, M. Lin, X. Li, A. G. Hauptmann, B. Raj. *Beyond gaussian pyramid: Multi-skip feature stacking for action recognition*, 2014, arXiv: 1411.6660.

DOI: 10.25209/2079-3316-2017-8-4-327-345

[11] P. Koniusz, Fei Yan, Ph.-H. Gosselin, K. Mikolajczyk. “Higher-Order Occurrence Pooling for Bags-of-Words: Visual Concept Detection”, *IEEE Transactions on Pattern Analysis & Machine Intelligence*, **39** (2017), pp. 313–326.

[12] G. J. Brostow, J. Shotton, J. Fauqueur, R. Cipolla. “Segmentation and recognition using structure from motion point clouds”, Computer Vision – ECCV 2008, Lecture Notes in Computer Science, vol. **5302**, Springer, Berlin–Heidelberg, 2008, pp. 44–57.

[13] V. Ramanathan, Sh. Mishra, P. Mitra. “Quadtree decomposition based extended vector space model for image retrieval”, 2011 IEEE Workshop on Applications of Computer Vision (WACV) (5–7 Jan. 2011, Kona, HI, USA), pp. 139–144.

[14] J. C. van Gemert, J. M. Geusebroek, C. J. Veenman, A. W. M. Smeulders. “Kernel codebooks for scene categorization”, Computer Vision – ECCV 2008, Lecture Notes in Computer Science, vol. **5304**, Springer, Berlin–Heidelberg, pp. 696–709.

[15] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong. “Locality-constrained linear coding for image classification”, 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (13–18 June 2010, San Francisco, CA, USA), 9 p.

[16] J. Sánchez, F. Perronnin, T. Mensink, J. Verbeek.. “Image Classification with the Fisher Vector: Theory and Practice”, *International Journal of Computer Vision*, **105**:3, pp. 222–245.

[17] K. Chatfield, K. Simonyan, A. Vedaldi, A. Zisserman. “Return of the devil in the details: Delving deep into convolutional nets”, British Machine Vision Conference BMVC 2014 (1–5 September, 2014, Nottingham, UK), URL: http://www.bmva.org/bmvc/2014/files/paper054.pdf

[18] G. Csurka, F. Perronnin. “Fisher Vectors: Beyond Bag-of-Visual-Words Image Representations”, VISIGRAPP 2010: Computer Vision, Imaging and Computer Graphics. Theory and Applications, Communications in Computer and Information Science, vol. **229**, Springer, Berlin–Heidelberg, pp. 28–42.

[19] Ch. Olah. *Understanding LSTM Networks*, August 27, 2015, URL: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

[20] Kh. Soomro, A. R. Zamir M. Shah. *UCF101: A Dataset of 101 Human Action Classes From Videos in The Wild*, CRCV-TR-12-01, 2012.

[21] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, M. Paluri. *C3D: generic features for video analysis*, 2014, arXiv: 1412.0767.

[22] J. Donahue, L.-A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, T. Darrell. *Long-term recurrent convolutional networks for visual recognition and description*, 2014, arXiv: 1411.4389.

Sample citation of this publication:

Aleksandr Buyko, Andrey Vinogradov. “Action recognition on video using recurrent neural networks”, *Program systems: Theory and applications*, 2017, **8**:4(35), pp. 327–345. (*In Russian*).

URL: http://psta.psiras.ru/read/psta2017_4_327-345.pdf