

Институт ИТАСУ

Кафедра инженерной кибернетики

Направление подготовки: 01.03.04 «прикладная математика»

Квалификация (степень): бакалавр

КУРСОВАЯ РАБОТА

учебная дисциплина

«Методы и средства обработки изображений»

Тема: Оценка близости кадров видео по цветовому профилю

VIII семестр 2020 у. г.

Студент: Малышковский О.В.

Группа: БПМ-16-2

Преподаватель: доц., к.т.н. Д. В. Полевой

Оценка:

Дата:

Оглавление

Актуальность темы.....	3
Формулировка содержательной постановки задачи.....	4
Формулировка математической постановки задачи	5
Рассмотрение алгоритма решения задачи	6
Выбор цветового пространства	6
Метод получения ключевых кадров.....	7
Критерии оценки близости кадров.....	8
Этапы решения задачи.....	10
Описание методов программы	12
Выводы.....	13
Список литературы.....	14

Актуальность темы

В связи с развитием технологий фото- и видеосъемки и в целом распространенности сети Интернет накоплено огромное количество мультимедийной информации доступной для просмотра. Этот контент часто используют в своей профессиональной деятельности дизайнеры. Например, для получения нужного визуального эффекта некоторого сайта может понадобиться набор видеороликов, у которых будет приблизительно одна цветовая гамма, чтобы они хорошо сочетались с общим оформлением веб-страницы.

В сети представлено очень большое количество видеоконтента, и для того, чтобы было возможно найти схожие по цветовым характеристикам видео, нужен алгоритм, который оценивал бы цветовую близость. Для решения данной задачи в работе используются цветовые гистограммы, для которые исследованы различные критерии оценки их сходства.

Цель работы: исследовать применимость различных критериев оценки близости кадров видео по цветовым гистограммам.

Формулировка содержательной постановки задачи

В работе необходимо продемонстрировать применимость оценки близости кадров разных видео, основывая лишь на их цветовой характеристике, представленной в виде гистограмм. В качестве исходных данных будут использоваться короткие цветные видеоролики (не более 15 секунд) в формате mp4. Ролики можно условно разделить на два кластера: в первом ролики в оттенках синего (снежная местность, волны воды, и др.), а во втором находятся лесные пейзажи в оттенках зеленого цвета.

Для того, чтобы ускорить вычисление сравниваться будут не все подряд кадры, а будут найдены key frames (ключевые кадры) для каждого видеоролика. Ключевые кадры – кадры из видео, по которым становится понятным то, что происходит на видео в целом. Перевод кадров в гистограммы также способствует уменьшению объема используемых данных для вычислений.

Формулировка математической постановки задачи

Пусть $f_1, f_2, \dots, f_N \in \mathbb{R}^{n \times m \times k}$ – множество изображений (кадров видео) размера $n \times m \times 3$, где 3 – число каналов у изображения.

Для каждого f_i построим гистограмму H_i :

$H_i = \{ H[0], H[1], \dots, H[r], \dots, H[n] \}$ – вектор, где элемент $H[r]$ представляет собой число значений цвета под индексом r в матрице изображения f_i одного канала.

$H'_i = \{ \frac{H[0]}{p}, \frac{H[1]}{p}, \dots, \frac{H[2]}{p} \}$ – нормированный вектор, где $p = n \times m$

Необходимо найти функцию, которая будет оценивать близость пары гистограмм:

$$(H_i, H_j): \langle H_i, H_j \rangle \rightarrow [0, 1]$$

где 1 означает полное сходство двух гистограмм, а 0 – что они совершенно разные

Рассмотрение алгоритма решения задачи

Выбор цветового пространства

В данной работе проводится сравнение близости кадров на двух цветовых пространствах.

RGB: наиболее распространенное цветовое пространство. Состоит из значения Red, Green и Blue, все в диапазоне от 0 до 255. Прозительно ярко-красный 255–0–0. 0–0–0 - черный, 255–255–255 - белый.

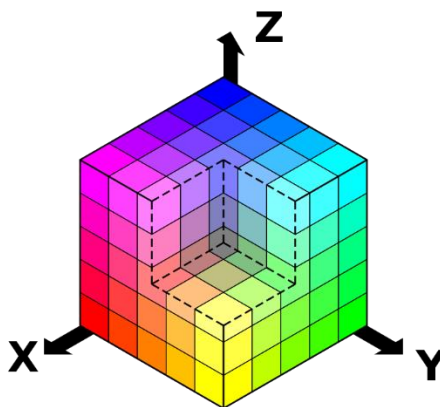


Рис.1 Пространство RGB в виде куба

HSV: это цветовое пространство понятно для человека. Hue, Saturation, Value — оттенок, насыщенность, яркость. Оттенок — это тон цвета, насыщенность - насколько он близок к серому, а яркость - к черному.

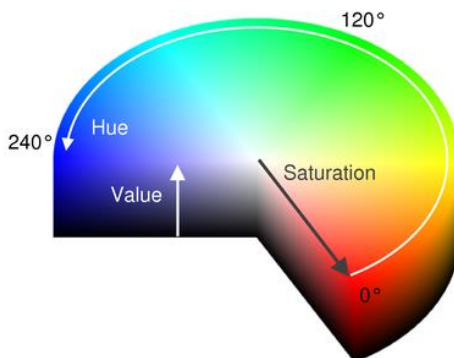


Рис.2 Пространство HSV в виде цилиндра

Метод получения ключевых кадров

Ключевые кадры - не что иное, как самая значимая часть всего видео, исключая избыточные кадры. В последнее десятилетие была проделана значительная работа в области обработки видео для разделения данного видео на отдельные кадры.

Сегментация кадров — это первый шаг выделения ключевого кадра, который в основном относится к обнаружению перехода между последовательными сценами. Основная идея большинства методов состоит в том, чтобы измерить и сравнить сходство между последовательными кадрами.

Так как для нашей задачи нет необходимости в очень хорошем выделении ключевых кадров, а главным является уменьшение объема данных для последующих операций, то можно использовать любой подход. Для детектирования будем считать корреляцию между гистограммами двух кадров. Как только её значение между предыдущим ключевым кадром и текущим кадром станет меньше 0,8 (этот порог выбран эмпирически), то текущий кадр зачислим в группу ключевых.



Рис.3 В верхней части представлены кадры видео, а внизу выделенные ключевые кадры

Критерии оценки близости кадров

Существуют множество функции близости для двух гистограмм. Для данной работы были выделены 4 такие функции:

Корреляция - основана на коэффициенте корреляции Пирсона, и обычно подходит, когда H_1 и H_2 могут быть интерпретированы как вероятностные распределения.

$$d(H_1, H_2) = \frac{\sum_i H'_1(i) * \sum_i H'_2(i)}{\sqrt{\sum_i H_1'^2(i) * \sum_i H_2'^2(i)}}$$

где $H'_k(i) = H_k(i) - N^{-1} \sum_j h_k(j)$, N - число цветов, h – значения гистограммы.

Для корреляции высокий балл представляет собой большее соответствие, чем низкий балл. Идеально совпадение равно 1, а максимальное несовпадение равно -1 ; значение 0 указывает на отсутствие корреляции.

Расстояние хи-квадрат - основано на распределении хи-квадрат, которая также указывает, коррелированы ли два распределения.

$$d(H_1, H_2) = \sum_i \frac{(H_1(i) - H_2(i))^2}{H_1(i) + H_2(i)}$$

Низкий балл представляет собой лучшее совпадение, чем высокий балл. Идеальное совпадение равен 0, а общее несоответствие не ограничено (в зависимости от размера гистограммы).

Интерсекция (пересечения) - основана на простом пересечении двух гистограмм. Это означает, что выясняется, что общего между этими двумя, и берется сумма по всем уровням гистограмм.

$$d(H_1, H_2) = \sum_i \min(H_1(i), H_2(i))$$

Для этого показателя высокие оценки указывают на хорошие совпадения, а низкие - на плохие. Если обе гистограммы нормированы до 1, то идеальное совпадение равно 1, а общее несоответствие равно 0.

Расстояние Бхаттачария (перекрытие) – также является мерой перекрытия двух распределений.

$$d(H_1, H_2) = \sqrt{1 - \frac{\sum_i H_1(i) * H_2(i)}{\sqrt{\sum_i H_1(i) * \sum_i H_2(i)}}}$$

В этом случае низкие оценки указывают на хорошие совпадения, а высокие оценки указывают на плохие совпадения. Идеальное совпадение - 0, а общее несовпадение - 1.

В целом, стоит нормализовать гистограммы перед сравнением их, потому что такие понятия, как пересечение гистограммы имеют мало смысла без нормализации.



Histograms:		Matching measures:			
Model:		Correlation:	Chi square:	Intersection:	Bhattacharyya:
					
Exact match:		1.0	0.0	1.0	0.0
					
Half match:		0.7	0.67	0.5	0.55
Total mis-match:		-1.0	2.0	0.0	1.0

Рис.4 Сравнение оценок, полученных описанными методами. Слева показаны разные случаи: гистограммы полностью совпадают, гистограммы похожи и гистограммы совсем разные.

Этапы решения задачи

Для выполнения всех операций по обработке видео и гистограмм отдельных кадров будем использовать библиотеку OpenCV. Для каждого видео получим вектор ключевых кадров. Для детектирования будем считать корреляцию между гистограммами двух кадров. Как только её значение между предыдущим ключевым кадром и текущим кадром станет меньше 0,8 (этот порог выбран эмпирически), то текущий кадр зачислим в группу ключевых.

Переводим ключевые кадры в пространство HSV или RGB. Для каждого ключевого кадра строим гистограмму. Для всех комбинаций пар, полученных для двух сравниваемых видео, вычислим 4 функции оценки близости кадров описанных ранее. Затем возьмем среди них минимальное, среднее и максимальное значения.

Проведем вычисления для двух схожих по цветовой гамме видео:

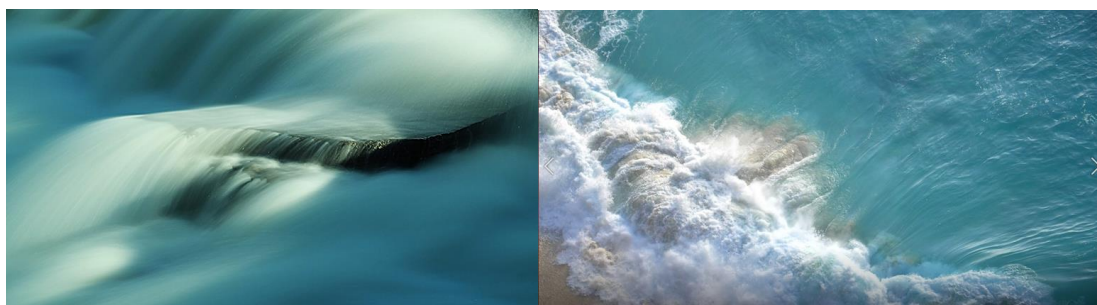


Рис.5 Слева и справа фрагменты из двух видеороликов

В цветовом пространстве RGB (найдено 7 и 6 ключевых кадров для двух видео):

Функция близости Значения	Корреляция	Хи-квадрат	Интерсекция	Расстояние Бхаттачария
Минимальное	0.100985	1393.12	2.96114	0.809615
Среднее	0.206157	28185.3	4.73764	0.841435
Максимальное	0.362129	69761.8	6.20513	0.87566

В цветовом пространстве HSV (найдено 2 и 6 ключевых кадров для двух видео):

Функция близости Значения	Корреляция	Хи-квадрат	Интерсекция	Расстояние Бхаттачария
Минимальное	0.0851556	1913.03	3.53289	0.814437
Среднее	0.187801	54707.7	5.21441	0.852981
Максимальное	0.353079	211249	6.78606	0.901783

Проведем вычисления для двух несхожих по цветовой гамме видео:

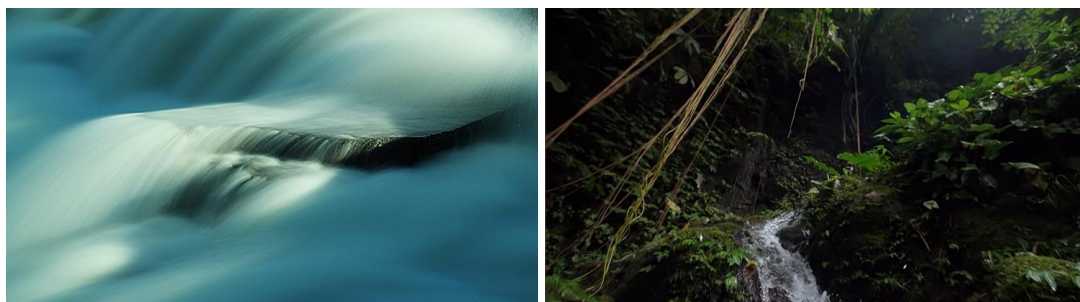


Рис.6 Слева и справа фрагменты из двух видеороликов

В цветовом пространстве RGB (найдено 7 и 13 ключевых кадров для двух видео):

Функция близости Значения	Корреляция	Хи-квадрат	Интерсекция	Расстояние Бхаттачария
Минимальное	0.00105646	387.101	0.184932	0.955149
Среднее	0.0178583	12075.5	0.382264	0.981984
Максимальное	0.0488215	34829.2	0.544353	0.96642

В цветовом пространстве HSV (найдено 2 и 8 ключевых кадров для двух видео):

Функция близости Значения	Корреляция	Хи-квадрат	Интерсекция	Расстояние Бхаттачария
Минимальное	-0.0110014	1461.49	0.0923733	0.969111
Среднее	-0.00576511	51619	0.340399	0.982597
Максимальное	0.00373648	250266	0.582361	0.996671

Расстояние Бхаттачария хоть и дает для похожих видео значения меньше, чем для разных, но даже 0,8 для схожих видео достаточно высокое значение, значит такой критерий плохо подходит для нашей задачи. За исключением расстояния Бхаттачария в HSV в среднем более адекватные результаты (хотя и в RGB тоже можно различить похожи или нет два видео).

Описание методов программы

Ссылка на хранилище: https://gitlab.com/malynkovsky/image_proc/-/tree/master/course_work

Видео были скачаны с ресурса: <https://www.videvo.net/stock-video-footage>

В данной работе были реализованы следующие методы:

C++

1. `get_hist_in_HSV` – вычисление гистограммы для кадра в пространстве HSV. На вход принимает матрицу (кадр).
2. `get_hist_in_BGR` – вычисление гистограммы для кадра в пространстве RGB. На вход принимает матрицу (кадр).
3. `compare_two_frames` – похожи ли два кадра (для поиска key-frames). На вход принимает две матрицы.
4. `compare_frames` – сравнение двух кадров по одной из метрик. похожи ли два кадра (для поиска key-frames). На вход принимает две матрицы, номер метрики, цветовое пространство.
5. `get_key_frames` – получение key-frames для видео. похожи ли два кадра (для поиска key-frames). На вход принимает путь к видеофайлу и цветовое пространство.
6. `get_min_max_avg` – вывод минимального, среднего и максимального значений метрики. На вход вектор значений посчитанных метрик.
7. `get_comparing_data` – сравнение key_frames для двух видео На вход принимает вектора с ключевыми кадрами для двух видео, номер метрики.
8. `full_compare_two_videos` – полное сравнение двух видео с помощью описанные выше методов. На вход принимает пути к сравниваемым видеофайлам и цветовое пространство.

Номер 0 – для пространства HSV, 1 – RGB. Номера метрик от 0 до 3 в порядке, в котором они описаны ранее.

В *main* продемонстрирована работа всех метрик для двух схожих и несхожих по цветовой гамме видеофайлов в двух цветовых пространствах.

Выводы

Согласно полученным результатам, можно сделать следующие выводы:

1. Существует много функций сходства для оценки близости кадров видео. Среди рассмотренных в работе хорошо проявили себя корреляция, расстояние хи-квадрат, интерсекция.
2. Результаты вычислений похожи в цветовых пространствах HSV и RGB, но в первом результаты немного лучше, так как оно более линейно к изменению цветов в отличии от второго.
3. Данный способ оценки близости кадров опирается лишь на цветовую составляющую и сможет найти видео в разных цветовых гаммах. Если же цвета похоже, но объекты на видео различаются, то метод оценки только по цветовому профилю это не различит.
4. При сравнении всех ключевых кадров одного видео со всеми ключевыми кадрами другого рассмотренный алгоритм не устойчив, если одно видео будет содержать сцены в разной цветовой гамме.

Список литературы

1. A. Kaehler, G. Bradski Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library. – O'Reilly Media, Inc, 2017.
2. A. V. Kumthekar, J. K. Key frame extraction using color histogram method. – Department of E&TC, Shivaji University, Kolhapur, 2013.