О кластеризации в анализе движения в изображении с неподвижной камеры

А.И. Шапошников, Е.В. Шапошникова
Томский институт академии ВЭГУ

В [1] представлен один из подходов к решению задачи кластеризации (разбиения) ряда изображений, образующих видеоряд, на участки одного цвета. Предлагаемый метод позволяет получать кластеры, пересекающиеся с несколькими кадрами. Однако кластеризация может быть проведена не всегда. В предлагаемой работе исследуются условия проводимости кластеризации, и решается проблема непрерывности для получаемых результатов.

Введение

Задача анализа изображения неподвижной камеры вызвана большим числом прикладных задач, например слежения, управления движением и др. В качестве основных этапов в этом анализе выделяются фильтрация изображения, выделение движущихся объектов и слежение за выделенными объектами – трекинг (tracking) [2]. Несмотря на значительные продвижения в каждой из этих задач в отдельности (отделение теней и бликов, выделение движущегося множества [3], трекинг выделенных объектов, анализ группы вращений и т.д.), задача анализа движений объектов произвольного ролика (например, из [4]) еще не решена. Как отмечено в [2], «практическое использование вычислительной техники ограничено, и исследования должны продолжаться».

В этой работе обсуждены техника кластеризации видеоизображений на основе Mean Shift и некоторые результаты ее применения

Фильтрация и кластеризация

Дадим некоторые определения. Неопределяемым понятием будем считать кадр видеоизображения, т.е. то, что мы видим на экране монитора. Последовательность кадров назовем видеорядом. В пределах этой статьи рассматриваем только кадры, снятые с одной и той же неподвижной камеры. Каждому кадру сопоставим его номер в последовательности. Пиксель – это элемент кадра видеоизображения, соответствующий некоторой точке на мониторе. Каждый пиксель отождест-

вим с точкой **(*x,y,t,r,g,b*)** в пространстве $\mathbf{R}^6$, где ***x,y*** – координаты точки на экране; ***t*** – номер кадра; ***r,g,b*** – компоненты цвета точки экрана.

Процедуру MeanShift-кластеризации рассмотрим, следуя [6]. Выбираем ***h*** – шаг окна (как правило, ***h*=16** вполне подходит) и для каждой точки **(*x,y,t,r,g,b*)** строим рекуррентно последовательность центроидов C_i. Рассмотрим результаты кластеризации двух последовательных кадров видеоролика, снятого с неподвижной камеры [7] (рис. 1).

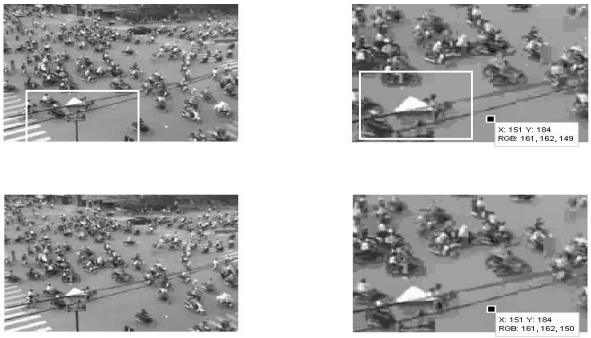

Рис. 1. Кластеризация кадров

Как видим, полотно дороги после кластеризации имеет на обоих кадрах одинаковый цвет (161, 162, 150). Согласованность касается и более тонких объектов (рис. 2, выделенный прямоугольник).

Рис. 2. Выделение объектов

Объект – мотоциклист движется, что заметно по изменению и первой координаты, и цвета брюк на исходных кадрах. На обработанных

кадрах брюки имеют разные координаты, но одинаковый цвет (153,155,144), т.е. они попали в единый кластер, расположенный в разных кадрах.

Влияние скоростей объектов

При проведении такой схемы кластеризации и фильтрации в один кластер могут попасть точки, принадлежащие совершенно разным объектам, которые имеют одинаковые координаты в разных кадрах.

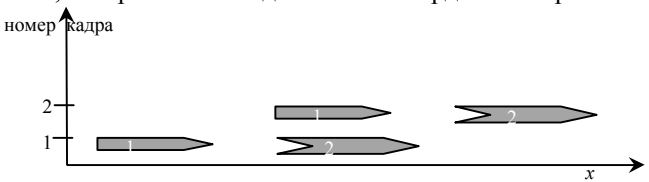

Рис. 3. Использование ограниченного числа кадров

Рассмотрим рис. 3. Здесь плоскость кадра представлена одной пространственной координатой ***x***. По вертикали откладывается номер кадра. Из-за низкой частоты кадров и высоких скоростей *1* и *2*, объект *1* оказался во втором кадре на том месте, на котором был объект *2*. Цвета объектов схожие, значит, существуют пиксели $\boldsymbol{p_1}$ и $\boldsymbol{p_2}$, принадлежащие разным объектам, которые будут классифицированы в один кластер при любом значении шага. Результат неправильный.

Чтобы обеспечить правильную кластеризацию, надо согласовать частоту кадров, масштаб по ***t*** и скорости наблюдаемых объектов.

При включении в видеоролик промежуточных кадров ситуация меняется (рис. 4).

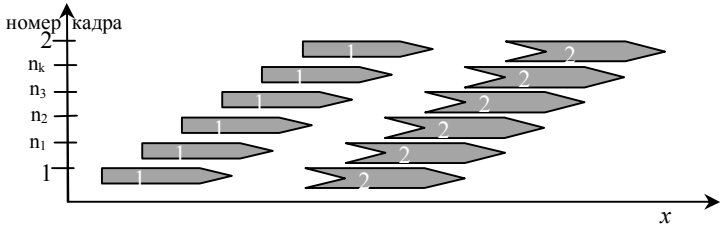

Рис. 4. Включение дополнительных промежуточных кадров

В этом случае даже без изменения первоначальных настроек мы получим правильный результат. При частоте ***n*** кадров в секунду получаем ограничение на скорости объектов в кадре не более ***nh*** пикселей

в секунду. Таким образом, мы получаем допустимое для кластеризации пикселей расположение объектов в кадрах.

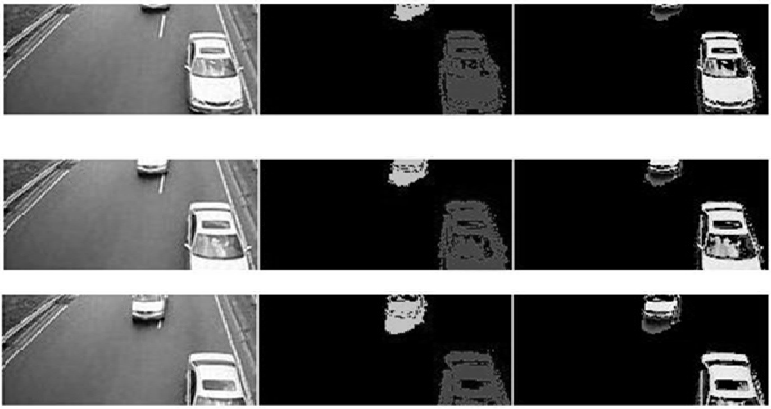

Рис. 5. Ролик с автомобилем

Соответствующий модуль был реализован на MatLab. Обрабатывался ролик размером 160×120 пикселей, 8 с, с легковыми машинами, движущимися в кадре. Ролик был снят камерой наблюдения, размещенной на автомобильной магистрали. Смещение автомобилей в соседних кадрах составляло не более 13 пикселей за кадр. Шаг окна ***h=16*** пикселей. Общее время обработки сюжета составило 354 с. Все автомобили были правильно кластеризованы и раскрашены каждый в свой цвет.

На рис. 5 показаны три последовательных кадра. Слева – исходное изображение. В центре – результат кластеризации. Как видим, пиксели корпуса каждого автомобиля на всех трех кадрах составляют единый кластер. Справа – те же кластеры без раскрашивания всех элементов кластера в единый цвет (пиксели имеют свои цвета).

Литература

1. **Шапошников А.И.** Выделение движущихся объектов в анализе изображения неподвижной камеры// Информационные технологии и математическое моделирование: Материалы VIII Всероссийской научно-практической конференции с международным участием (13–4 ноября 2009 г.). Томск: Изд-во Том. ун-та, 2009. Ч. 2. С. 297-301.

2. **Cucchiara R., Grana C., Piccardi M., Prati A.** Detecting moving objects, ghosts, and shadows in video streams // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2003. Vol. 25, № 10. P. 1337–1342.
3. **Гаганов В., Конушин А.** Сегментация движущихся объектов в видеопотоке. http://cgm.computergraphics.ru/content/view/67.
4. http://YouTube.com.
5. **Comaniciu D., Meer P.** Mean Shift Analysis and Applications // IEEE Int. Conf. Computer Vision (ICCV'99). Kerkyra, Greece, 1999. P. 1197-1203.
6. http://youtube.com/watch?v=gbPLV2kB4-Y&feature=related