

Стрельников К.Н., Солдатов С.А., Ватолин Д.С. "Качественное определение глобального движения кадры использованием векторов движения", материалы девятого научно-практического семинара "Новые информационные технологии в автоматизированных системах", стр. 47-55, Москва, март 2006

Качественное определение глобального движения кадры с использованием векторов движения

Стрельников К.Н., Солдатов С.А., Ватолин Д.С.

МГУ им. Ломоносова, факультет Вычислительной Математики и Кибернетики,
лаборатория Компьютерной Графики
{kstrelnikov, stass, dmitriy}@graphics.cs.msu.ru

Аннотация

В статье предлагается новый алгоритм определения глобального движения кадры, описываемого с помощью четырех параметров. Алгоритм основывается на векторах движения. Существенный недостаток векторов движения – низкая надежность. Поэтому дополнительно предлагается способ фильтрации ошибочных векторов движения.

1 Введение

Задача определения глобального движения кадры (Global Motion) исследуется давно. Существующие алгоритмы применяются как для сжатия [1], так и для обработки видео. Например, в задачах построения панорамного изображения [2] или стабилизации видео с дрожанием кадры [3]. Алгоритмы определения глобального движения кадры можно разделить на три группы:

- 1) использующие для работы особые точки (Feature);
- 2) использующие для работы вектора движения (Motion Vector);
- 3) глобальный поиск.

Алгоритмы из каждой группы имеют характерные сильные и слабые стороны.

Использование аппарата особых точек требует значительных временных затрат на выбор особых точек (Feature Selection) и на их отслеживание (Feature Tracking) [4]. Преимуществом же данного подхода является высокая надежность определения

смещений. Как пример использования особых точек для определения глобального движения, можно привести алгоритм 2D Ridge Motion [3]. Надежность определения смещений при использовании векторов движения гораздо ниже, чем при использовании особых точек. Но у алгоритмов определения векторов движения (Motion Estimation) [5] есть большое преимущество – это скорость. Алгоритмы, использующие для определения глобального движения вектора движения, описаны в статьях [3,6]. Под алгоритмами глобального поиска мы понимаем алгоритмы с кадрами целиком без применения промежуточных преобразований. Такой алгоритм описан в статье [7].

В этой статье мы предлагаем алгоритм высококачественного определения глобального движения использующий вектора движения. Алгоритм использует сильную сторону векторов движения (скорость) и минимизирует погрешность определения глобального движения, связанную с низкой надежностью векторов движения.

2 Фильтрация ошибочных векторов

Аппарат векторов движения (Motion Estimation) давно и очень активно используется для решения задач связанных с обработкой и компрессией видео. Основное преимущество этого подхода перед другими – его простота. Существенная помеха при использовании векторов движения заключается в наличии ошибочных векторов, их особенно часто можно видеть на монотонных областях кадра. Пример ошибочных векторов движения можно видеть на рисунке 1 (они обведены).



Рисунок 1. Вектора движения для 3-его кадра из видео последовательности «flower»

Каждому вектору движения можно поставить в соответствие величину, характеризующую степень соответствия действительному движению, то есть его достоверность. Эту величину можно использовать для фильтрации поля векторов, отбрасывать вектора с низкой достоверностью.

Нами предлагается следующий подход для оценки достоверности вектора движения. Достоверность вектора движения для блока в позиции (x, y) определяется исходя из трех характеристик:

- $error(x, y)$ – ошибка приближения (сумма абсолютных разностей пикселей блока в позиции (x, y) и сопоставленного ему блока из предыдущего кадра);
- $disp(x, y)$ – дисперсия блока в позиции (x, y) ;
- $dev(x, y)$ – среднеквадратичное отклонение вектора от четырех соседних векторов:

$$dev_{x,y} = \frac{1}{4} \cdot \sum_{\substack{i=-1;1 \\ j=-1;1}} (mv(x, y)^x - mv(x, y)^x)^2 + (mv(x, y)^y - mv(x, y)^y)^2$$

$mv(x, y)^x, mv(x, y)^y$ – проекции вектора движения.

Мы предлагаем функцию доверия векторам движения, задаваемую формулой (1), которая позволяет получить адекватное значение достоверности вектора (*belief*).

$$belief(x, y) = \left(a \cdot error(x, y) + \frac{b}{disp(x, y)^2} + c \cdot dev(x, y) \right)^{-1} \quad (1)$$

a, b, c – параметры.

Опытным путем было установлено, что наилучший результат достигается при следующих значения параметров: $a=0.25$; $b=32$; $c=1$.

Малым значения функции доверия соответствуют менее достоверным векторам, а вектора, для которых значение функции доверия максимально, с наибольшей точностью соответствуют действительному движению.

На рисунке 2 приведен пример определения достоверности векторов движения. Степень достоверности убывает от темно синих областей к зеленым, далее к желтым и красным. Можно видеть, что вектора, соответствующие красным областям на изображении функции доверия, некорректно определяют движение.

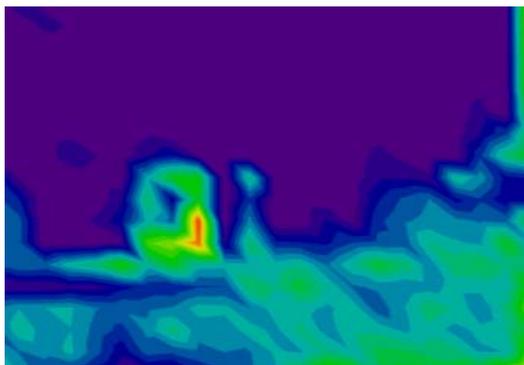
Наша задача минимизировать погрешность, связанную с недостатками векторов движения, поэтому далее будем использовать только вектора для которых $belief(x, y) > T$ (некоторого порога).



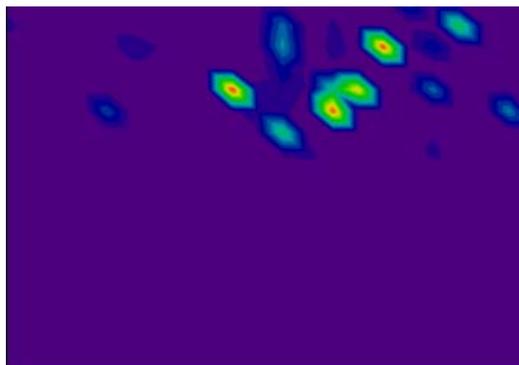
кадр с векторами движения



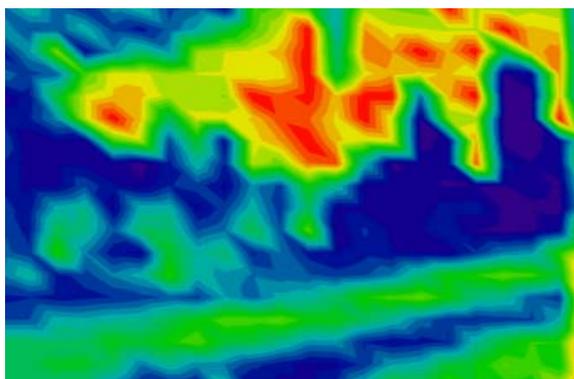
ошибка приближения блоков



дисперсии блоков



среднеквадратичное отклонение



функция доверия векторам движения

Рисунок 2. Определение достоверности векторов движения

3 Определение глобального движения кадра

Имея информацию о движении фрагментов кадра в виде векторов движения, можно определить, как двигался весь кадр в целом. Глобальное движение кадра можно аппроксимировать разными моделями. Мы определяем глобальное движение между кадрами I_1 и I_2 , как преобразование вида (2).

$$\begin{pmatrix} x^2 \\ y^2 \end{pmatrix} \sim \begin{pmatrix} s \cdot \cos \varphi & -s \cdot \sin \varphi \\ s \cdot \sin \varphi & s \cdot \cos \varphi \end{pmatrix} \begin{pmatrix} x^1 \\ y^1 \end{pmatrix} + \begin{pmatrix} a \\ b \end{pmatrix} = F(x^1, y^1; a, b, s, \varphi) \quad (2)$$

Где $(x^1; y^1)$ – точки кадра I_1 , а $(x^2; y^2)$ – соответствующие точки кадра I_2 . a, b, s, ϕ – параметры модели: сдвиг по горизонтали, сдвиг по вертикали, увеличение и поворот. Задача определения глобального движения сводится к минимизации функции от четырех параметров (3).

$$F(a, b, s, \phi) = \sum_{i,j} \left[\left(s \cdot \cos \phi \cdot x^1_{i,j} - s \cdot \sin \phi \cdot y^1_{i,j} + a - x^2_{i,j} \right)^2 + \left(s \cdot \sin \phi \cdot x^1_{i,j} + s \cdot \cos \phi \cdot y^1_{i,j} + b - y^2_{i,j} \right)^2 \right] \quad (3)$$

3.1 Определение увеличения

Важно максимально точно определить значение параметра s , так как он может вносить наибольшую погрешность в модель. Опишем наш алгоритм определения значения параметра s . Заметим, что для двух векторов задача об определении параметра s решается очень просто.

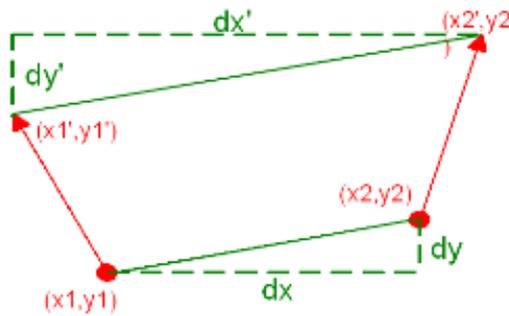


Рисунок 3. Определение параметра s для двух векторов

Возьмем два вектора, приведенных на рисунке 3. Для них справедлива формула (4).

$$\begin{pmatrix} x_i' \\ y_i' \end{pmatrix} = \begin{pmatrix} s_{1,2} \cdot \cos \phi & -s_{1,2} \cdot \sin \phi \\ s_{1,2} \cdot \sin \phi & s_{1,2} \cdot \cos \phi \end{pmatrix} \cdot \begin{pmatrix} x_i \\ y_i \end{pmatrix} + \begin{pmatrix} a \\ b \end{pmatrix} \quad i = 1, 2 \quad (4)$$

Проведя несложные вычисления, получаем формулу (5), определяющую значение параметра s .

$$s_{1,2} = \sqrt{\frac{(x_2' - x_1')^2 + (y_2' - y_1')^2}{(x_2 - x_1)^2 + (y_2 - y_1)^2}} \quad (5)$$

Далее из множества достоверных векторов выбираем N произвольных. Для каждого выбранного вектора V_n ($n \in 1..N$) определяем параметр z_n , который задает значение увеличения для конкретного вектора движения. Для этого из множества достоверных

векторов выбираем произвольные M векторов mv_m ($m \in 1..M$). Используя формулу (5) вычисляем набор значений $s_{n,m}$ для векторов V_n и mv_m , где $m \in 1..M$. И определяем z_n по формуле (6), как медиану от набора значений $s_{n,m}$.

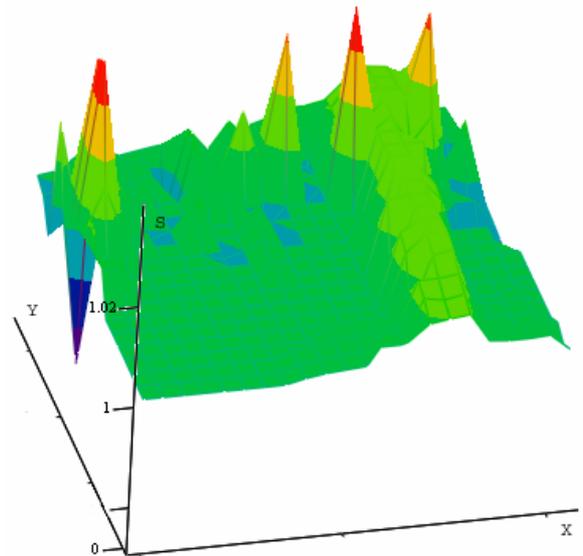
$$z_n = \underset{m \in 1..M}{\text{median}} (s_{n,m}) \quad n \in 1..N \quad (6)$$

Получив значения параметров z_n ($n \in 1..N$), определяем s аналогично формуле (6), как медиану от набора значений z_n .

Некоторые вектора могут соответствовать движущимся или более близким к камере объектам, как на рисунке 4. Их нельзя использовать при определении движения кадра, так как они будут вносить ошибку.



кадр с векторами движения



параметр z для всех векторов

Рисунок 4. Дополнительная фильтрация векторов

Такие вектора можно отбросить, проведя фильтрацию по найденным для них значениям z_n . Фильтрация осуществляется следующим образом: из множества векторов V_n ($n \in 1..N$) оставляем 90% векторов VF_k ($k \in 1..K, K < N$) для которых значение параметра z мало отличается от значения параметра s для всего кадра.

3.2 Определение угла поворота и сдвигов вдоль осей координат

Теперь определим оставшиеся параметры модели. Для этого минимизируем значение функции потерь $F(a, b, \varphi)$, задаваемой формулой (7).

$$F(a,b,\varphi) = \sum_{k \in [1;K]} \left[\left(s \cdot \cos \varphi \cdot VF^{X1}_k - s \cdot \sin \varphi \cdot VF^{Y1}_k + a - VF^{X2}_k \right)^2 + \left(s \cdot \sin \varphi \cdot VF^{X1}_k + s \cdot \cos \varphi \cdot VF^{Y1}_k + b - VF^{Y2}_k \right)^2 \right] \quad (7)$$

- VF^{X1}, VF^{Y1} – координаты начала вектора;
- VF^{X2}, VF^{Y2} – координаты конца вектора.

Для этого нужно продифференцировать выражение (7) по каждому параметру, приравнять результат к нулю и решить полученную систему из трех уравнений.

В результате, получим выражения (8-10) для определения параметров a, b и φ .

$$\varphi = \arccos \left(K \cdot \sum_{k \in [1;K]} VF^{X2}_k{}^2 + K \cdot \sum_{k \in [1;K]} VF^{Y2}_k{}^2 - \left[\sum_{k \in [1;K]} VF^{X2}_k \right]^2 - \left[\sum_{k \in [1;K]} VF^{Y2}_k \right]^2 \right) \quad (8)$$

$$\arccos \left(s \cdot \left[\sum_{k \in [1;K]} VF^{X1}_k \cdot \sum_{k \in [1;K]} VF^{X2}_k + \sum_{k \in [1;K]} VF^{X1}_k \cdot \sum_{k \in [1;K]} VF^{Y2}_k - \sum_{k \in [1;K]} VF^{Y1}_k \cdot \sum_{k \in [1;K]} VF^{X2}_k + \sum_{k \in [1;K]} VF^{Y1}_k \cdot \sum_{k \in [1;K]} VF^{Y2}_k - \right. \right. \\ \left. \left. - K \cdot \sum_{k \in [1;K]} VF^{X1}_k \cdot VF^{X2}_k - K \cdot \sum_{k \in [1;K]} VF^{X1}_k \cdot VF^{Y2}_k + K \cdot \sum_{k \in [1;K]} VF^{X1}_k \cdot VF^{Y2}_k - K \cdot \sum_{k \in [1;K]} VF^{Y1}_k \cdot VF^{Y2}_k \right] \right)$$

$$a = \frac{1}{K} \cdot \left(\sum_{k \in [1;K]} VF^{X2}_k - s \cdot \cos \varphi \cdot \sum_{k \in [1;K]} VF^{X1}_k + s \cdot \sin \varphi \cdot \sum_{k \in [1;K]} VF^{Y1}_k \right) \quad (9)$$

$$b = \frac{1}{K} \cdot \left(\sum_{k \in [1;K]} VF^{Y2}_k - s \cdot \sin \varphi \cdot \sum_{k \in [1;K]} VF^{Y1}_k - s \cdot \cos \varphi \cdot \sum_{k \in [1;K]} VF^{X1}_k \right) \quad (10)$$

На рисунке 5 представлена полная схема работы алгоритма.

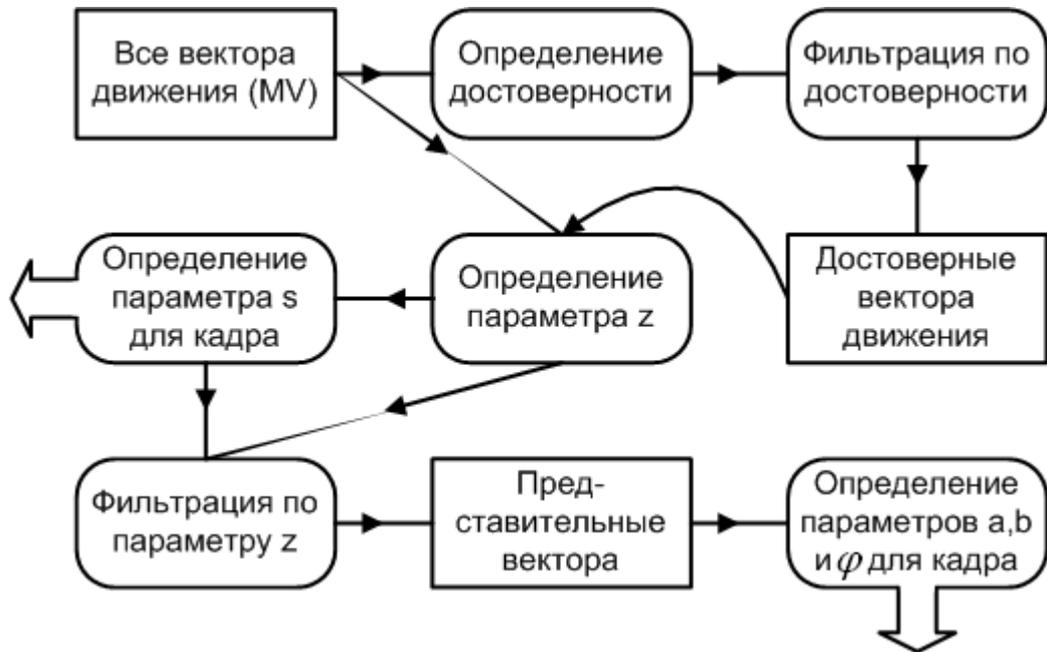


Рисунок 5. Схема алгоритма определения глобального движения

4 Результаты

Сравним предложенный нами алгоритм с определения параметров предложенной модели движения и метод наименьших квадратов без предварительной фильтрации ошибочных векторов. Для сравнения качества работы, приведем примеры панорамы, построенной по найденным параметрам движения. Алгоритм построения панорамы использовался один. Из рисунка 6 видно, что предложенный алгоритм значительно превосходит конкурента по точности определения параметров движения.



метод наименьших квадратов



предложенный алгоритм

Рисунок 6. Сравнение алгоритмов

5 Заключение

Предложенный алгоритм с высокой точностью определяет глобальное движение кадра, описываемое моделью (2). Также достигнута высокая скорость работы. Алгоритм позволяет обрабатывать 60 кадров видео формата 352x288 в секунду на Pentium 4 1.6GHz. Алгоритм можно с успехом применить при решении соответствующих задач. Например, для построения панорамного изображения или стабилизации видео.

6 Список литературы

- [1] Y.Keller, A.Averbuch. “Fast Global Motion Estimation for MPEG-4 Video Compression”. PACKET VIDEO 2003, April 2003.
- [2] A.Bartoli, N.Dalal, B.Bose, R.Horaud. “From Video Sequences to Motion Panoramas”. IEEE Workshop on Motion and Video Computing, December 2002, pp. 201-207.
- [3] Kwong Kin Wah Eric. “Video Stabilization Algorithms”. ELEC 533 Project Report, spring 2003
- [4] Shi, J., Tomasi, C. “Good Features to Track”. IEEE Conference on Computer Vision and Pattern Recognition, pp. 593-600, 1999
- [5] G. de Haan, “Progress in motion estimation for video format conversion”. IEEE Transactions on Consumer Electronics, Vol. 46, No. 3, Aug. 2000, pp. 449-459.
- [6] Irani M., Rousso B., Peleg S. “Recovery of Ego-Motion Using Image Stabilization”. Computer Vision and Pattern Recognition '94, pp. 454-460, 1994
- [7] A.Litvin, J.Konrad, W.C.Karl. “Probabilistic Video Stabilization Using Kalman Filtering and Mosaicking”. Proceedings of SPIE Conference on Electronic Imaging, 2003.