

УДК 004.932.2

А.А. Молчанов, В.И. Кортунюв

Национальный аэрокосмический университет имени Н.Е. Жуковского «ХАИ», Харьков

МЕТОД ОЦЕНКИ ДВИЖЕНИЯ ОПТИЧЕСКОГО ПОТОКА С ВЗВЕШИВАНИЕМ ИЗМЕРЕНИЙ БЛОКОВ ИЗОБРАЖЕНИЯ

В статье предложен метод оценки движения видеокамеры или платформы, использующийся в обработке изображений для задач определения параметров движения. Предложен метод определения и формализации оптического потока с взвешиванием измерений для блоков изображений. На основе данных численного моделирования для тестовых изображений показана эффективность предложенного метода. Описанные в работе методы и алгоритмы являются достаточно универсальными и находят практическое применение в самых различных областях.

Ключевые слова: оптический поток, вектор движения, поле движения, метод сопоставления блоков, алгоритм Lucas-Kanade, метод наименьших квадратов, Sum of Absolute Differences.

Введение

В настоящее время оценка параметров движения широко применяется в процессах сжатия и обработки видео. Оценка движения играет важную роль в таких задачах, как преобразование частоты кадра, сегментизация видео (выделение и сопровождение объектов в кадре), преобразование развертки в прогрессивную, шумоподавление, восстановление сжатого видео, повышение качества видео, определение движения по последовательности видеок кадров и во многих других [1]. Движение объектов перед камерой или движение камеры в неподвижной окружающей обстановке приводит к соответствующим изменениям на изображении и это изменение можно использовать для восстановления относительного движения. Оптическим потоком называется кажущееся изменение яркостной картинки в форме скорости, которые движутся вместе с наблюдаемыми объектами. В идеале оптический поток соответствует полю движения, при условии, что объекты не изменяют энергетическую освещенность на плоскости изображений в процессе движения в сцене и это позволит нам оценивать относительное движение, исходя из изменения изображения во времени. В [2] предложен метод оценки движения на основе векторов кандидатов, но данный метод имеет ошибки определения векторов движения в однородных, а также гладких областях.

Достоинством предлагаемого метода является перебор сокращенного числа векторов, и тем самым повышенная вычислительная эффективность. Однако это достоинство нивелируется в случае использования постоянного размера блоков и областей поиска. Тем не менее, при фиксированном размере блоков повышается устойчивость векторов в гладких областях, поскольку вероятность попа-

дания контрастных деталей возрастает вместе с ростом площади блока. Это позволяет частично решить одну из основных проблем блочных методов оценки движения – проблему поиска векторов в гладких, а также слабо текстурированных областях. Для блоков из таких областей мера близости текущего и предыдущего кадров принимает очень близкие значения (или даже равные, в случае абсолютно гладких областей) для различных векторов движения, т.е. эффективность использования ее значений в качестве основного критерия выбора вектора сводится к минимуму. Повысить вероятность успешного нахождения вектора движения позволяет использование блоков большего размера, но это эффективно лишь в случаях, когда размер блока больше размера гладких областей.

Можно выделить несколько основных групп методов определения оптического потока: дифференциальные методы оценки, метод фазовой корреляции (phase correlation), метод сопоставления блоков (block matching algorithms) [3].

Как показано в [4], наиболее широко на практике применяются методы сопоставления блоков. Это обусловлено универсальностью, невысокой вычислительной сложностью и высокой эффективностью, а также из-за простоты аппаратной реализации. Общая процедура работы этих методов такова:

а) текущий кадр разбивается на множество непересекающихся блоков;

б) для каждого блока текущего кадра производится поиск наиболее похожего блока в предыдущем кадре;

в) разность между позициями текущего и предыдущего блока называется вектором движения текущего блока.

В данной работе предложен метод определения движения, использующие адаптивно меняю-

щийся размер и адаптивную стратегию поиска вектора движения, зависящий от структуры блока. Описанная ниже процедура метода совпадает с общей процедурой работы, каждый из этапов которой подробно описан.

Метод формирования и анализа оптического потока

Пусть $I(x, y, t)$ – освещенность ПЗС матрицы в точке изображения (x, y) в момент времени t . Тогда, если $u(x, y)$ и $v(x, y)$ – x, y -компоненты вектора оптического потока в этой точке, можно ожидать, что освещенность будет той же самой в момент $t + \Delta t$ в точке $(x + \Delta x, y + \Delta y)$, где $\Delta x = u\Delta t$ и $\Delta y = v\Delta t$. Иными словами, $I(x + u\Delta t, y + v\Delta t, t + \Delta t) = I(x, y, t)$ для малого интервала времени Δt . Этого единственного ограничения недостаточно для однозначного определения двух неизвестных u и v . Ясно также, что нам хотелось бы воспользоваться тем, что поле движения почти везде непрерывно. Если яркость изменяется гладко по (x, y) и t , то мы можем разложить левую часть уравнения в ряд Тейлора и отсюда получить

$$I(x + \Delta x, y + \Delta y, t + \Delta t) = I(x, y, t) + \Delta x \frac{\partial I}{\partial x} + \Delta y \frac{\partial I}{\partial y} + \Delta t \frac{\partial I}{\partial t} + e,$$

где e содержит члены высших порядков малости по $\Delta x, \Delta y$ и Δt , начиная со второго. Сократим $I(x, y, t)$ и разделим на Δt , тогда получим

$$\frac{\partial I}{\partial x} \frac{dx}{dt} + \frac{\partial I}{\partial y} \frac{dy}{dt} + \frac{\partial I}{\partial t} \approx 0. \quad (1)$$

Коротко это уравнение можно написать в виде $\frac{dI}{dt} = 0$, где в левой части стоит полная производная

I по времени. Введя обозначение $u = \frac{dx}{dt}, v = \frac{dy}{dt}$, запишем (1) в виде

$$\frac{\partial I}{\partial x} u + \frac{\partial I}{\partial y} v + \frac{\partial I}{\partial t} = 0. \quad (2)$$

Это уравнение называют уравнением оптического потока [5], в котором две неизвестные компоненты вектора движения для каждого пикселя. Воспользуемся предположением о близости векторов движения для всех пикселей из окрестности точки (x, y) , вектор движения (u, v) одинаковый, получим большее количество уравнений для двух неизвестных.

Для упрощения расчетов предлагается использовать поле движения, формируемое на основе оценок смещений не точек, а отдельных блоков, на которые предварительно разбивается изображение.

Каждый кадр видео последовательности размером $M \times N$ пикселей разбивается на множество не-

перекрывающихся блоков B_{ij} размером $n \times n$ пикселей, где i, j – координаты центра блока.

$$B_{ij}(t) = \{I(x_i, y_j, t) : x_k, y_l \in B_{ij}\}, \\ i - n/2 < k < i + n/2; \\ j - n/2 < l < j + n/2.$$

Блоки размещены последовательно слева направо, сверху вниз. Разбиение производится так, что все блоки покрывают весь кадр, т.е. их суммарная площадь равна площади кадра. Задача определения движения сводится к задаче поиска векторов движения $(u, v)_{ij}$ для каждого блока I_{ij} .

Обозначим вектор движения $\vec{z} = (u, v)$. Тогда в момент времени $t + \Delta t$ получим яркостную функцию для блока B :

$$B(t + \Delta t) = I(x + u, y + v, t + \Delta t).$$

Запишем оптимальную оценку параметров вектора движения центра блока, как оптимальную задачу

$$\hat{z} = \min_z \sum_{y, x \in B} |B(t) - B(t + \Delta t)|^2. \quad (3)$$

Оценка (3) не может быть записана в явной форме, а получается численно в итерационной процедуре минимизации критерия

$$Q(z) = \sum_{y, x \in B} |B(t) - B(t + \Delta t)|^2. \quad (4)$$

В нашем случае возможно преобразование переменных, делающее модельную функцию линейной, разложив $B(t + \Delta t)$ в ряд Тейлора.

Продифференцировав функцию $Q(z)$ по u, v и приравняв частные производные к нулю, получим систему уравнений для определения значений оценок параметров (u, v) :

$$\frac{\partial Q(z)}{\partial u} = 0, \quad \frac{\partial Q(z)}{\partial v} = 0.$$

Дифференцируя (4), получим:

$$\begin{cases} \sum_{x, y \in B} \left[(B(t) - B(t + \Delta t)) \left(\frac{dB^0(t + \Delta t)}{dx} \right) \right] = 0; \\ \sum_{x, y \in B} \left[(B(t) - B(t + \Delta t)) \left(\frac{dB^0(t + \Delta t)}{dy} \right) \right] = 0, \end{cases}$$

где

$$B(t + \Delta t) = B^0(t + \Delta t) + \frac{dB^0(t + \Delta t)}{dx} u + \frac{dB^0(t + \Delta t)}{dy} v.$$

Оптимальный вектор МНК-оценки параметра \hat{z} определен на основе следующего векторно-матричного выражения:

$$\hat{z} = (A^T A)^{-1} \cdot A^T \Delta B,$$

где A – вектор-строка данных

$$A = \begin{bmatrix} \frac{dB^0(t + \Delta t, x, y)}{dx} \Big|_{\substack{x=x_i \\ y=y_j}} & \frac{dB^0(t + \Delta t, x, y)}{dy} \Big|_{\substack{x=x_i \\ y=y_j}} \end{bmatrix},$$

ΔB – вектор-столбец правых частей уравнений

$$\Delta B = B(t) - B^0(t + \Delta t);$$

$$\Delta B = \overline{\Delta B} + \varepsilon,$$

где ε – вектор-столбец ошибки модели;

$\overline{\Delta B}$ – вектор-столбец истинных значений y .

Оценки \hat{z} являются случайными величинами. Их можно представить как сумму истинного значения z и некоторой случайной ошибки Δz :

$$\hat{z} = z + \Delta z;$$

$$\hat{z} = z + (A^T A)^{-1} A^T \cdot \varepsilon,$$

где $(A^T A)^{-1} A^T \cdot \varepsilon$ – вектор ошибок оценок параметров \hat{z} .

Общепринятой в статистике мерой точности является дисперсия оценок модели. Точность оценок, их эффективность определяется характером их выборочного распределения, и в частности, мерой из случайного разброса относительно истинных значений оцениваемых параметров. Мера случайного разброса значений оценок \hat{z} относительно истинных значений z определяется дисперсией и ковариационной матрицей. Дисперсию оценки \hat{z} можно найти как значения диагональных элементов ковариационной матрицы

$$\text{cov}(\hat{z}) = \begin{pmatrix} \sigma_1^2 & \text{cov}(z_1 z_2) & \dots & \text{cov}(z_1 z_n) \\ \text{cov}(z_2 z_1) & \sigma_2^2 & \dots & \text{cov}(z_2 z_n) \\ \dots & \dots & \dots & \dots \\ \text{cov}(z_n z_1) & \text{cov}(z_n z_2) & \dots & \sigma_n^2 \end{pmatrix},$$

где $\sigma^2 = D(\hat{z})$ – дисперсия оценки.

Выражение ковариационной матрицы оценок \hat{z} можно представить [6]:

$$\text{cov}(\hat{z}) = \sigma_\varepsilon^2 \cdot (A^T A)^{-1}.$$

Матрица $A^T A$, в дальнейшем обозначим G , тесно связана с ковариационной матрицей, которая является исходным объектом для различных видов многомерного анализа. Дисперсии и ковариации оценок \hat{z} могут быть определены как элементы матрицы $\sigma_\varepsilon^2 \cdot (A^T A)^{-1}$ [7].

Алгоритм Lucas-Kanade основан на поиске особых точек изображения [8]. Поиск особых точек происходит с помощью матрицы Гессе, детерминант которой достигает экстремума в точках максимального изменения градиента яркости. Он хорошо детектирует углы и края линий объектов на изображении.

Матрица данных G является произведением матриц Якоби, и согласно выражению

$$M\{\nabla B^T \nabla B\} = -M\{\nabla^2 B\} \quad [9],$$

можно утверждать, что в асимптотическом смысле анализ матрицы Гессе соответствует анализу произведения матриц Якоби.

Предполагается анализ на основе числа обусловленности матрицы G , который дает более объективную оценку. Число обусловленности матрицы данных G , является мерой чувствительности системы линейных уравнений к погрешностям задания вектора правых частей уравнений, и вычисляется по формуле [10]:

$$\text{cond}(G) = \|G^{-1}\| \cdot \|G\|,$$

где $\|G\|$ – норма матрицы G ;

G^{-1} – обратная матрица G .

Чем больше число обусловленности cond , тем сильнее воздействие и тем более неустойчив процесс нахождения решения. Хотя число обусловленности матрицы зависит от выбора нормы, если матрица хорошо обусловлена, то её число обусловленности будет мало при любом выборе нормы, а если она плохо обусловлена, то её число обусловленности будет велико при любом выборе нормы. В работе [11] показано, что устойчивая модель имеет число обусловленности $\text{cond}\{1 \dots 100\}$. При cond от 1 до 10 модель имеет хорошую вычислительную устойчивость, а от 10 до 100 – удовлетворительную вычислительную устойчивость. При cond больше 100 модель имеет неудовлетворительную чувствительность.

Для исследования рассмотрена последовательность видео кадров, состоящая из двух изображений (рис. 1).

Анализ видеопоследовательности представляет собой распределение, где каждому блоку рис. 1, б соответствует показатель текстуры, число обусловленности матрицы G . График анализа текущего кадра (рис. 2) позволит определить относительное расположение блоков друг относительно друга и выделить не информационные блоки.

Вектор движения определен в областях с числом обусловленности от 1 до 10, в областях при cond больше 10, блоки считаются ложно обнаруженными и игнорируются (вектор движения равен 0), в гладких областях определитель матрицы данных нулевой, cond равен бесконечности, вектор движения не вычисляется для этих областей.

Введем показатель, который будет характеризовать изображение или фрагменты изображения, относительно оценки числа обусловленности матрицы G :

$$k = \frac{1}{NM} \sum_{ij} \frac{1}{\text{cond}(G_{ij})},$$

где NM – размер области.

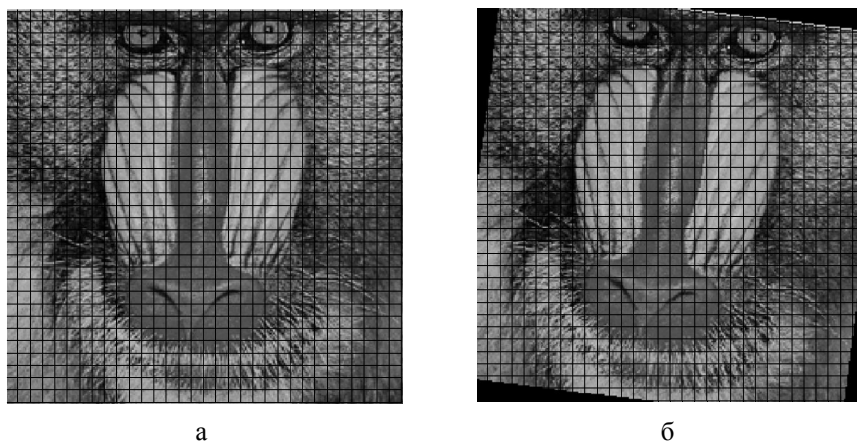


Рис. 1. Последовательность изображений:
а – предыдущий кадр, б – текущий кадр

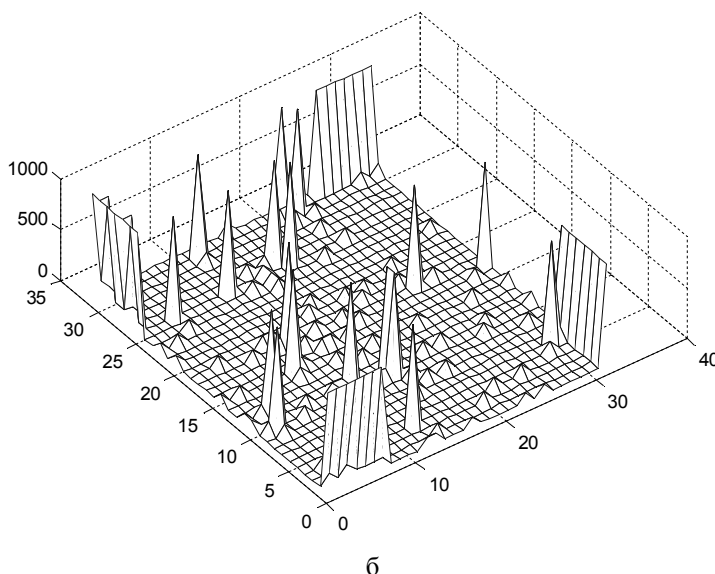
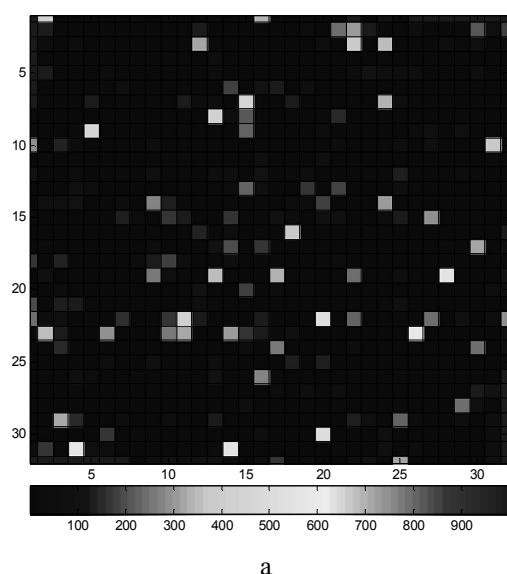


Рис. 2. Анализ текущего кадра

Таким образом, вычислив величину k для области, становится возможным адаптивно определить порог для ошибок векторов, в случае не превышения которого блок необходимо разбить на блоки меньшего размера.

Блоки малого размера используются только в тех областях кадра, где это необходимо. Применяем минимальный размер блока 8×8 пикселей.

Разбиение блока на четыре подблока производится адаптивно:

- а) в областях со сложной текстурой;
- б) в областях, где текущий размер блока не может обеспечить приемлемый уровень числа обусловленности.

Метод блочной оценки движения

Задача определения движения сводится к задаче поиска вектора движения каждого блока B_{ij} [12]. При этом вектора $(u, v)_{ij}$ определяются соотношением:

$$(u, v)_{ij} = \arg \min_{(u, v)_{ij} \in B_{ij}} (\text{SAD}(t, i, j, (u, v)_{ij})),$$

где $\text{SAD}(t, i, j, (u, v)_{ij})$ – (Sum of Absolute Differences), функция соответствия блоков, мера близости блоков текущего и предыдущего кадров. Функция SAD использует для сравнения текущий блок и соответствующую область в последующем блоке, принимает абсолютную разность между каждым пикселем.

Эти различия суммируются для создания простой метрики сходства.

Для каждого блока текущего кадра производится минимизация функции соответствия блоков по 4-му аргументу, при этом область минимизации может быть любой, единственным ограничением является то, что она должна быть подмножеством области поиска.

В качестве вектора движения для каждого блока выбирается аргумент минимума функции соот-

ветствия, вычисленный в этом блоке. Фактически при вычислении функции соответствия производится определение похожести двух блоков: блока текущего кадра и блока предыдущего кадра, смещенного на вектор $(u, v)_{ij} \cdot \Delta t$.

Таким образом, процесс минимизации функции соответствия является поиском блока предыдущего кадра, наиболее похожего на текущий блок.

Базовыми подходами для вычисления блочного определения движения являются [13]:
полный перебор,

шаблонные методы,
метод иерархического поиска;
методы, использующие вектора-кандидаты.

Мы предлагаем использовать комбинированный метод, основываясь на методе полного перебора используя иерархический поиск. Алгоритм иерархического поиска заключается в следующем.

Перед началом поиска производится вычисление $N-1$ уменьшенных «копий» текущего и опорного кадров, при этом каждая очередная копия в $2n$ (n – натуральное число) раз меньше предыдущей (рис. 3).



Рис. 3. Схема алгоритма иерархического поиска

Пары кадров одинакового размера будем называть уровнями, т.е. на одном уровне опорный и текущий кадры одинакового размера. Тогда все множество пар кадров можно представить N уровнями.

Пронумеруем уровни согласно размеру содержащихся в них кадров от меньшего к большему: 1-й уровень будет содержать кадры минимального размера, N -й – кадры исходного размера. Процесс оценки движения состоит из N итераций, на каждой из которых обрабатывается пара кадров из уровня с соответствующим номером, т.е. обработка идет от кадров меньшего размера к большему. На каждой итерации производится ОД, при этом в качестве стартовой точки на каждой итерации выбирается векторное поле, полученное с предыдущей итерации.

Другими словами, каждая очередная итерация производит уточнение векторов, вычисленных на предыдущей итерации. При переходе на очередную итерацию размеры области поиска и блоков, для которых оцениваются вектора, обычно увеличиваются в $2n$ раз для того, чтобы число блоков в кадре на каждой итерации не менялось.

Дополнительным преимуществом данного метода является устойчивость к шуму, поскольку во время уменьшения изображений, как правило, удаляются высокочастотные шумы.

Однако вместе с высокочастотными шумами могут пропасть и мелкие детали, что приведет к неправильному определению движения в детализированных областях.

На рис. 4 приведен пример применения предлагаемого алгоритма для определения движения из последовательности изображений рис. 1.

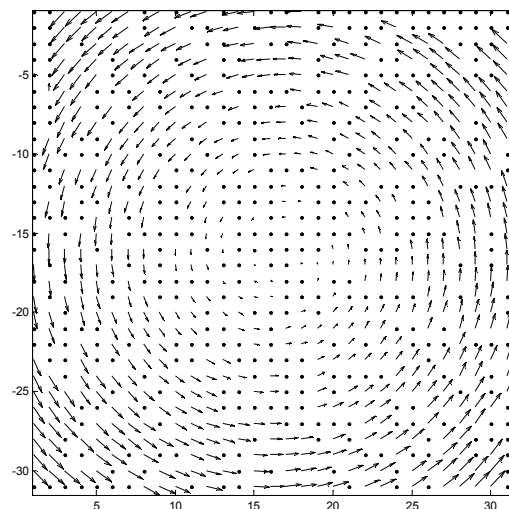


Рис. 4. Поле векторов текущего кадра

Обзор методов использования оптического потока для определения динамических параметров движения камеры описан в работе [14].

Заключение

Предложен метод определения и формирования оптического потока с взвешиванием блоков изображений, а также использование оптического потока в задаче определения параметров движения по оптическому потоку.

Блочные алгоритмы оценки движения являются выгодным компромиссом по соотношению вычислительная сложность/точность найденных векторов. Комбинирование приемов из алгоритмов различной категории в рамках класса блочных методов позволяет построить универсальные алгоритмы оценки движения, обладающие заданным свойством и легко реализуемые аппаратно.

Рассмотренные методы обработки изображения могут применяться как еще одно средство автономной коррекции инерциальными навигационными системами в дополнение к спутниковым навигационным системам.

На основе данных численного моделирования для тестовых изображений, показана эффективность предложенного метода.

Список литературы

1. Обзор блочных методов оценки движения в цифровых видеосигналах [Текст] / С.В. Гришин, Д.С. Ватолин, А.С. Лукин, С.Ю. Путилин, К.Н. Стрельников // Программные системы и инструменты. Изд-во факультета ВМиК МГУ. – 2008. – № 9. – С. 50-62.
2. Hosur P.I. Motion vector field adaptive fast motion estimation [Текст] / P.I. Hosur, K.K. Ma // Presented at the Second Int. Conf. Inf., Commun., Signal Process. – Singapore, 1999. – P. 422-241.
3. Цифровая обработка изображений в информационных системах [Текст]: учеб. пособие / И.С. Грузман, В.С. Киричук и др. – Новосибирск: Изд-во НГТУ, 2002. – 352 с.
4. Путилин С. Быстрый алгоритм нахождения движения в видеопоследовательностях [Текст] / С. Путилин // Труды конференции Graphicon. – Новосибирск, 2006. – С. 407-410.
5. Смирнов Н. В. Курс теории вероятностей и математической статистики [Текст] / Н.В. Смирнов, И.В. Дунин-Барковский. – М.: Наука, 1965. – 512 с.
6. Вержбицкий В.М. Численные методы (линейная алгебра и нелинейные уравнения): учеб. пос. для вузов [Текст] / В.М. Вержбицкий. – М.: Высш. шк., 2000. – 266 с.

7. Айвазян С.А. Прикладная статистика: Исследование зависимостей [Текст] / С.А. Айвазян, И.С. Енюков, Л.Д. Мешалкин; Под ред. С.А. Айвазян. – М.: Финансы и статистика, 1985. – 487 с.

8. Цыпкин Я.З. Основы информационной теории идентификации [Текст] / Я.З. Цыпкин. – М.: Наука: Главная редакция физико-математической литературы, 1984. – 320 с.

9. Jing Xuan. An Efficient Three-Step Search Algorithm for Block Motion Estimation [Текст] / Xuan Jing, Lap-Pui Chau // IEEE Transactions on Multimedia. – 2004. – Vol. 6, N. 3 – P. 124-138.

10. Дикусар В.В. Некоторые численные методы решения линейных алгебраических уравнений [Текст] / В.В. Дикусар // Соросовский образовательный журнал. – 1998. – № 9. – С. 111-120.

11. Lucas B.D. An Iterative Image Registration Technique with an Application to Stereo Vision / B.D. Lucas, T. Kanade // Proceedings of the 7th international joint conference on Artificial intelligence. – 1981. – Vol. 2. – P. 674-679.

12. Лукьяница А.А. Цифровая обработка видеоизображений / А.А. Лукьяница, А.Г. Шишкин. – М.: Ай-Эс-Эс Пресс, 2009. – 518 с.

13. Ahn Tae Gyoung. Fast Full-Search Motion Estimation Based on Multilevel Successive Elimination Algorithm [Текст] / Tae Gyoung Ahn, Yong Ho Moon, Jae Ho Kim // IEEE Transactions on Circuits and Systems for Video Technology. – 2004. – Vol. 14, N. 11. – С. 202-218.

14. Молчанов А.А. Обзор методов определения параметров движения по данным оптического потока [Текст] / А.А. Молчанов, В.И. Кортунюв // Радіоелектронні і комп'ютерні системи : науково-технічний журнал. – 2013. – №2. – С. 80-85.

Поступила в редколлегию 28.01.2015

Рецензент: д-р техн. наук, проф. В.А. Лукин, Национальный аэрокосмический университет им. Н.Е. Жуковского «ХАИ», Харьков.

МЕТОД ОЦІНКИ РУХУ ОПТИЧНОГО ПОТОКУ ЗІ ЗВАЖУВАННЯМ ВИМІРЮВАНЬ БЛОКІВ ЗОБРАЖЕННЯ

А.О. Молчанов, В.І. Кортунюв

У статті запропоновано метод оцінки руху відеокамери або платформи, що використовується в обробці зображення для задач визначення параметрів руху. Запропоновано метод визначення та формування оптичного потоку з зважуванням вимірювань для блоків зображень. На основі даних чисельного моделювання для тестових зображень показана ефективність запропонованого методу. Описані в роботі методи та алгоритми є досить універсальними і знаходять практичне застосування в самих різних областях.

Ключові слова: оптичний потік, вектор руху, поле руху, метод зіставлення блоків, алгоритм Lucas-Kanade, Sum of Absolute Differences.

OPTICAL FLOW MOTION ESTIMATION METHOD BASED ON WEIGHTED IMAGING UNIT MEASUREMENT

A.O. Molchanov, V.I. Kortunov

In this paper a method for estimating the motion camcorder or platform used in image processing tasks to determine the motion parameters is proposed. A method for determining and shaping optical flow weighted measurements for image blocks is proposed. The effectiveness of the proposed method is shown on the basis of numerical simulation data for test images. The methods and algorithms described in this paper are versatile enough and are implemented in various fields.

Keywords: optical flow, motion vector, field motion, block matching algorithm, SAD, the nonlinear least squares method.