

КОНТУРНОЕ КОДИРОВАНИЕ ИЗОБРАЖЕНИЙ

Рассмотрены методы кодирования цифровых изображений с использованием центрально-симметричных масок, приводящие к образованию контурных изображений. Показана возможность прямого декодирования (деконволюции) контурного изображения. Предложен алгоритм быстрого сглаживания (интерполяции) локальных дефектов изображения, с применением итерационного метода верхней релаксации, при решении задачи Дирихле. Показаны особенности кодирования и декодирования изображения на гексагональной сетке.

Ключевые слова: деконволюция, прямоугольная и гексагональная сетки, оператор Лапласа, обработка изображения, верхняя релаксация, задача Дирихле

Введение

Кодирование изображений может преследовать различные цели, например, защиту визуальной информации от перехвата или сокращение ее избыточности, при передаче по каналам связи. При кодировании цифрового изображения производится его преобразование – фильтрация без потерь или с потерей некоторой части визуальной информации. Обычно потери допускаются с учетом того, что конечный приемник информации (телезритель, робот) не ощущает этой потери, поскольку она находится за пределом его восприятия. В технических приложениях для сокращения избыточности информации используются различные методы сжатия (компрессии) изображений и число этих методов весьма велико [1]. Среди них, обычно, отсутствуют методы, связанные с выделением контуров объектов изображения. Наоборот, методы выделения контуров с целью улучшения восприятия или распознавания изображений широко используются в системах технического зрения [2,3]. Другая ситуация в биологии. Например, зрительная система человека устроена таким образом, что в зрительную кору его головного мозга из каждого глаза поступает контурное изображение объектов наблюдения. Это обстоятельство обусловлено тем, что отдельные волокна зрительного нерва имеют центрально-симметричную on-off организацию своих рецептивных полей [4]. Математическое описание рецептивных полей дано, например, в работе [5]. В случае зрительной системы, контурное изображение, по-видимому, позволяет ускорить реакцию организма на контур – форму объектов наблюдения, не отягощенную избыточной информацией об уровнях освещения их поверхностей. Кроме того, переход к контурам позволяет свести до минимума энергетические затраты организма в процессе передачи информации от глаза к зрительной коре, поскольку в этом случае средняя частота импульсов действия снижается на несколько порядков.

В принципе, контурное изображение может быть полностью восстановлено до исходного, при известных значениях исходного изображения на его границах. Восстановление (декодирование) контурного изображения отличается простотой алгоритмов, что особенно важно при использовании методов параллельной обработки изображений для получения максимального эффекта от их использования.

Выделение контура

1	-2	1	-1	-1	-1	0	-1	0
-2	4	-2	-1	8	-1	-1	4	-1
1	-2	1	-1	-1	-1	0	-1	0
	а			б			в	

Рис.1 Маски фильтров выделения контуров, а) оператор двумерной дискретной производной второго порядка, б) однородно-периферийный оператор, в) оператор Лапласа.

Контур объекта цифрового изображения может быть выделен с помощью свертки изображения с некоторой маской (ядром, керном, матрицей, оператором, весовой функцией). В зависимости от вида маски и содержания изображения контур может иметь различный профиль (поперечное сечение). Часто используются квадратные маски различных размеров: 2x2, 3x3, 5x5 и др. На рис. 1 приведен пример масок размером 3x3. Здесь цветом выделены положительные (оттенки красного), отрицательные (оттенок синего) и нулевые (оттенок зеленого) значения весов ячеек маски. Сумма весов каждой маски равна нулю.

Маска на рис. 1 а) получена как результат последовательного применения прямой и обратной конечных разностей как в вертикальном, так и в горизонтальном направлениях. Маска на рис. 1 б) получена произвольно, заданием однородной периферии. Маска на рис. 1 в) получена как сумма применений второй конечной разности в вертикальном и горизонтальном направлениях и по аналогии со случаем непрерывных производных второго порядка называется оператором Лапласа.

Покажем вариант получения первой маски (рис 1а). Обозначим символом $v_{i,j}$ значение элемента изображения из i -той строки в вертикальном направлении и j -того столбца в горизонтальном направлении изображения, тогда обратная конечная разность элементов в вертикальном направлении определяется выражением:

$$f_{i,j} = v_{i,j} - v_{i-1,j} \quad (1)$$

Прямая конечная разность элементов в вертикальном направлении:

$$g_{i,j} = f_{i+1,j} - f_{i,j} \quad (2)$$

Вторая конечная разность элементов в вертикальном направлении определяется подстановкой выражения (1) в выражение (2):

$$g_{i,j} = v_{i-1,j} - 2v_{i,j} + v_{i+1,j} \quad (3)$$

Обратная конечная разность элементов в горизонтальном направлении определяется выражением:

$$h_{i,j} = g_{i,j} - g_{i,j-1} \quad (4)$$

Прямая конечная разность элементов в горизонтальном направлении:

$$e_{i,j} = h_{i,j+1} - h_{i,j} \quad (5)$$

Вторая конечная разность элементов в горизонтальном направлении определяется подстановкой выражения (4) в выражение (5):

$$e_{i,j} = g_{i,j-1} - 2g_{i,j} + v_{i,j+1} \quad (6)$$

Подстановкой выражения (3) в выражение (6) после приведения подобных членов и переноса начала отсчета в правый нижний угол получим выражение для второй конечной разности:

$$E_{i,j} = v_{i-2,j-2} + v_{i-2,j} + v_{i,j-2} + v_{i,j} + 4v_{i-1,j-1} - \\ - 2v_{i-1,j-2} - 2v_{i-1,j} - 2v_{i-2,j-1} - 2v_{i,j-1} \quad (7)$$

которое соответствует маске, изображенной на рис. 1а. Заметим, что в выражении (1), при вычислении элемента с начальным значением индекса i возникает трудность, если неизвестно значение элемента с индексом $i-1$. Аналогично в выражении (2) для конечного значения индекса i требуется значение элемента с индексом $i+1$. Та же ситуация с расчетами по индексу j . Трудность преодолевается введением понятия о границе изображения и требования вести расчет для неграничных элементов. При этом значения граничных элементов считаются известными. Нумерацию строк и столбцов изображения будем начинать с его левого верхнего угла с цифры «1». В выражении (7) будем считать границей две верхние строки и два левых столбца, а расчет вести для элементов в интервалах номеров: $3 \leq i \leq M$ и $3 \leq j \leq N$, где M и N число строк и столбцов изображения соответственно.

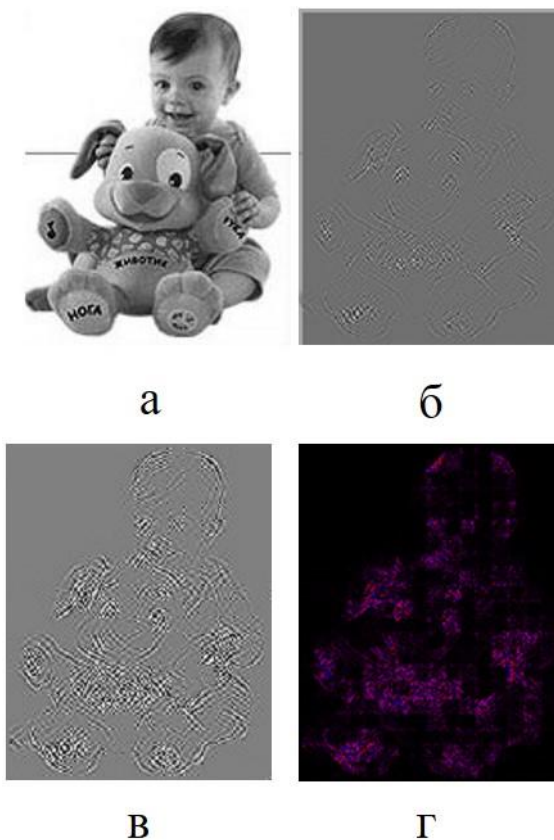


Рис.2 Изображения: а) исходное – v , б) контурное масштабированное – E , в) искусственно контрастированное – E , г) псевдо-цветное контурное – E .

Произведем свертку тестового изображения v с маской рис. 1 а), описываемую выражением (7). На рис. 2 приведены изображения: а) тестовое – v , б) масштабированное контурное – E , в) контрастированное контурное с обрезкой модуля значений пикселей больше 120 – E и масштабированное и г) контурное в псевдоцветах с красными положительными и синими отрицательными значениями контуров – E . Масштабирование в интервал $0 - 255$ позволяет отобразить отрицательные значения контуров в виде их темных аналогов, а положительные в светлых. Можно видеть на рис. 2 б), что масштабированное изображение состоит из пикселей, яркость которых в основном близка к среднему серому значению, поэтому для наглядности на рис. 2 в) приведен его искусственно контрастированный (искаженный по яркости) аналог, а на рис. г) отрицательные значения контуров инвертированы в положительные и поэтому контура на этом рисунке в черно-белом варианте изображения двойные однотонные.

Отметим особенность формирования маски а) рис. 1, которая состоит в том, что конечные разности вида (1), (2), (4), (5) можно брать в любом порядке, что не меняет вида выражения (7).

Произведем свертку тестового изображения v с маской рис. 1 б), описываемую выражением:

$$K_{i,j} = 8v_{i-1,j-1} - v_{i-2,j-2} - v_{i-2,j} - v_{i,j-2} - v_{i,j} - v_{i-1,j-2} - v_{i-1,j} - v_{i-2,j-1} - v_{i,j-1} \quad (8)$$

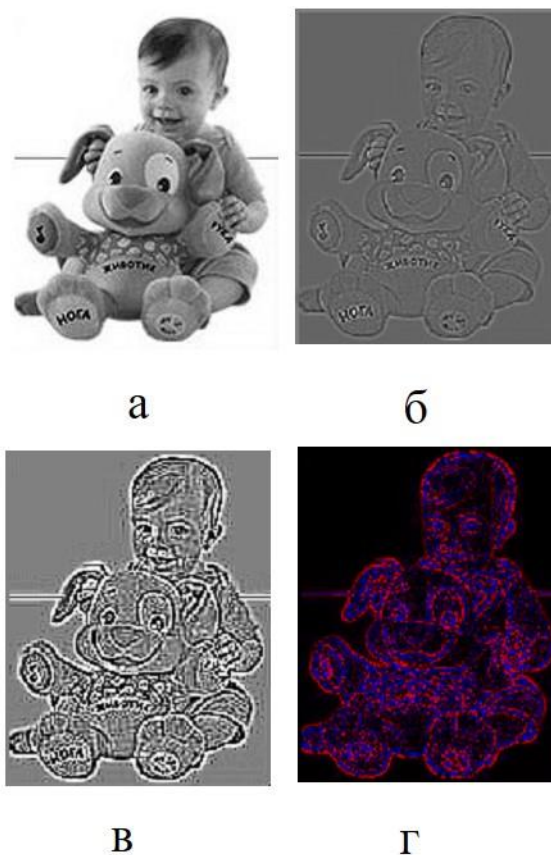


Рис.3 Изображения: а) исходное – v , б) контурное масштабированное – K , в) искусственно контрастированное – K , г) псевдо-цветное контурное – K .

На рис. 3 приведены изображения: а) тестовое – v , б) масштабированное контурное – K , в) контрастированное контурное с обрезкой модуля значений пикселей больше 120 – K и масштабированное и г) контурное в псевдоцветах с красными положительными и синими

отрицательными значениями контуров – К. Все изображения на рис. 3 получены аналогично изображениям на рис.2, но с другой маской.

Маски рис.1 а) и б) иногда называют операторами Лапласа из-за их центральной симметрии, чего явно недостаточно для такого названия. Наоборот, маска на рис. 1 в) является дискретным аналогом оператора Лапласа:

$$\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \quad (9)$$

Это выражение является суммой коэффициентов матрицы (маски), полученной суммированием двух матриц:

$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 2 & 0 \\ 0 & -1 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \end{pmatrix}, \quad (10)$$

где первое слагаемое соответствует выражению (3), а второе выражению (6).

Математическое выражение для маски рис. 1в) содержит всего 5 членов, поскольку угловые коэффициенты этой маски равны нулю и имеет вид:

$$L_{i,j} = 4v_{i,j-1} - v_{i-1,j-1} - v_{i,j-2} - v_{i,j} - v_{i+1,j-1}, \quad (11)$$

Равенство нулю правого нижнего элемента этой маски не позволяет в отличии от масок Е и К принимать этот элемент за начало отсчета ее элементов. – i, j . В качестве начала здесь выбран правый средний элемент, а расчет следует вести для элементов в интервалах номеров: $2 \leq i \leq M-1$ и $3 \leq j \leq N$, где: М и N число строк и столбцов изображения соответственно.

На рис. 4 приведены изображения: а) тестовое – v , б) масштабированное контурное – L, в) контрастированное контурное с обрезкой модуля значений пикселей больше 120 – L и масштабированное и г) контурное в псевдоцветах с красными положительными и синими отрицательными значениями контуров – L. Все изображения на рис. 4 получены аналогично изображениям на рис.2 и 3, но с другой маской и другой верхней границей.

Не смотря на различия полученных контурных изображений, все они могут быть восстановлены до изображения исходного вида, при известных границах этого изображения. Другими словами, при контурном кодировании изображения следует оставлять без изменения некоторые его границы. В рассмотренных случаях масок размером 3 x 3 ширина этих границ не превышает двух строк (столбцов), а сами границы не замкнуты. Для других масок толщина границ может достигать размеров маски. В любом случае границы должны быть такими, чтобы обеспечить вычисление начального элемента исходного изображения.

Восстановление изображения

Найти алгоритм восстановления (декодирования) из контурного – E изображения исходного изображения $v = Ve$, можно используя уравнение (7). Из которого следует, что:

$$\begin{aligned} Ve_{i,j} = & 2Ve_{i-2,j-1} + 2Ve_{i-1,j-2} + 2Ve_{i,j-1} + 2Ve_{i-1,j} + \\ & + E_{i,j} - 4Ve_{i-1,j-1} - Ve_{i-2,j-2} - Ve_{i-2,j} - Ve_{i,j-2} \end{aligned} \quad (12)$$

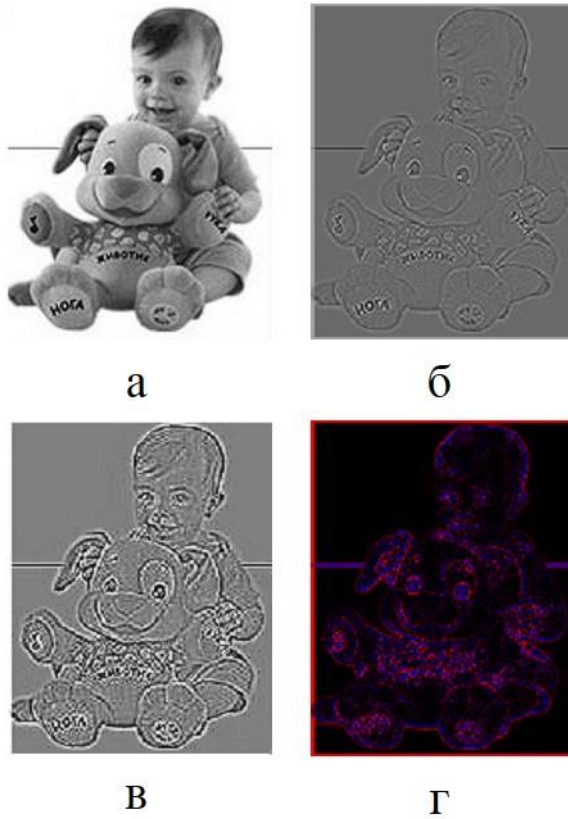


Рис.4 Изображения: а) исходное – v , б) контурное масштабированное – L , в) искусственно контрастированное – L , г) псевдо-цветное контурное – L .

При этом известны граничные значения:

$$Ve_{1,j} = E_{1,j} = v_{1,j}, \quad Ve_{2,j} = E_{2,j} = v_{2,j},$$

$$Ve_{i,1} = E_{i,1} = v_{i,1}, \quad Ve_{i,2} = E_{i,2} = v_{i,2}$$

и расчет ведется для элементов исходного изображения в интервале номеров:

$$3 \leq i \leq M \text{ и } 3 \leq j \leq N.$$

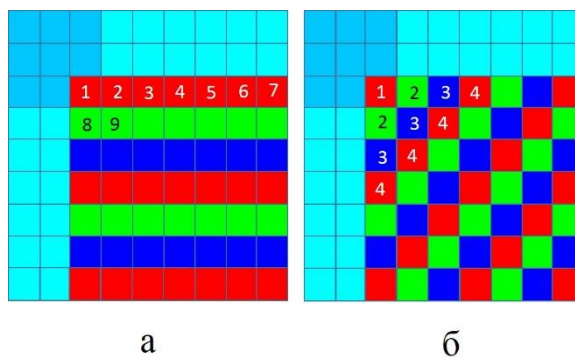


Рис.5 Прямоугольная решетка. Восстановление исходного изображения размера 9×9 из контурного того же размера: а) последовательно, б) параллельно.

Для иллюстрации последовательности декодирования контурного изображения на рис. 5 а) показан порядок восстановления небольшого исходного изображения при $M = 9$ и $N = 9$. Здесь восстановление начинается с элемента, обозначенного номером «1» и продолжается вдоль третьей строки изображения в порядке следования обозначенных номеров элементов. Вторая строка восстановления начинается с номера «8», затем «9» и так далее. Вторая и последующие строки восстанавливаются в аналогичном приведенному порядку. При этом восстанавливаемая маска, как и кодирующая движется вдоль строки касаясь нижним правым углом пронумерованного элемента изображения. На рисунке отдельные строки выделены различными (тремя) цветами (тонами). При описанном порядке восстановления изображения произойдет за $(M-2) \times (N-2)$ шагов.

Возможен другой порядок восстановления если имеется возможность параллельной обработки изображения. В этом случае контурное изображение декодируется целой строкой за один шаг обработки. На рис. 5 б) такие строки показаны одним цветом (тоном) и пронумерованы одинаковыми числами. Как видим эти строки расположены по диагоналям клетчатого поля изображения. При параллельном восстановлении оно произойдет всего за $M-2$ шага, если $M > N$ и за $N-2$ шага если $M < N$. Очевиден выигрыш во времени при параллельной обработке контурного изображения, что особенно важно, когда таких изображений много.

Алгоритм восстановления (декодирования) из контурного – K изображения исходного изображения $v = Vk$, можно получить используя уравнение (8). Из которого следует, что:

$$Vk_{i,j} = 8Vk_{i-1,j-1} - Vk_{i-2,j-2} - Vk_{i-2,j} - Vk_{i,j-2} - \\ - Vk_{i-2,j-1} - Vk_{i-1,j-2} - Vk_{i,j-1} - Vk_{i-1,j} - K_{i,j} \quad (13)$$

При этом известны граничные значения:

$$Vk_{1,j} = K_{1,j} = v_{1,j}, \quad Vk_{2,j} = K_{2,j} = v_{2,j},$$

$$Vk_{i,1} = K_{i,1} = v_{i,1}, \quad Vk_{i,2} = K_{i,2} = v_{i,2}$$

и расчет ведется для элементов исходного изображения в интервале номеров:

$$3 \leq i \leq M \text{ и } 3 \leq j \leq N.$$

Таким образом, порядок восстановления из K – контурного изображения отличается от предыдущего случая E – изображения только алгоритмом.

Алгоритм восстановления (декодирования) из контурного – L изображения исходного изображения $v = Vl$, можно используя уравнение (11). Из которого следует, что:

$$Vl_{i,j} = 4Vl_{i,j-1} - Vl_{i-1,j-1} - Vl_{i,j-2} - Vl_{i+1,j-1} - L_{i,j} \quad (14)$$

При этом известны граничные значения:

$$Vl_{1,j} = L_{1,j} = v_{1,j}, \quad Vl_{M,j} = L_{M,j} = v_{M,j},$$

$$Vl_{i,1} = L_{i,1} = v_{i,1}, \quad Vl_{i,2} = L_{i,2} = v_{i,2}$$

и расчет ведется для элементов исходного изображения в интервале номеров:

$$2 \leq i \leq M-1 \text{ и } 3 \leq j \leq N.$$

Заметим, что в выражении (14) присутствует член с индексом $i+1$, в отличие от предыдущих выражений (12) и (13), что обязывает правильно организовать циклы вычислений. Цикл по индексу i должен быть внутренним. Например, в системе Mathcad следует использовать программный модуль:

$$V_l := \left\{ \begin{array}{l} V_l \leftarrow L \\ \text{for } j \in 3..N \\ \quad \text{for } i \in 2..M-1 \\ \quad \quad V_{l,i,j} \leftarrow 4 \cdot V_{l,i,j-1} - V_{l,i-1,j-1} - V_{l,i,j-2} - V_{l,i+1,j-1} - L_{i,j} \end{array} \right. \quad (15)$$

Восстановление исходного изображения из контурного, полученного с применением любой из приведенных на рис. 1 масок по приведенным алгоритмам, позволяет получить точную копию исходного изображения, приведенного в наших примерах на рисунках 2, 3 и 4 с обозначением – **а**.

Гексагональная решетка

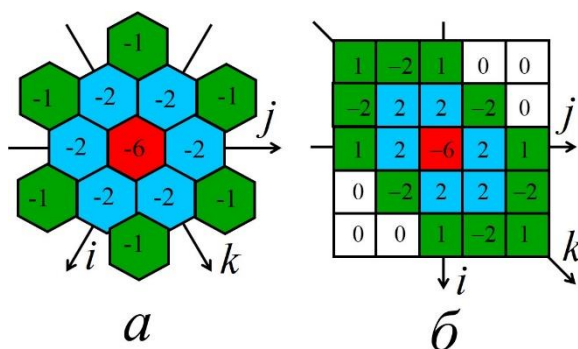


Рис.6: а) гексагональная маска дискретного аналога трехмерной производной второго порядка – «снежинка», б) двумерная карта «снежинки».

Гексагональные решетки для представления изображений обладают рядом преимуществ по сравнению с квадратными решетками пикселей. В частности, обладая более равномерным соединением пикселей в решетке, они обеспечивают лучшее угловое разрешение благодаря наличию большего числа ближайших соседей. Они лучше (более гладко) представляют контуры объектов [6, 7]. Ввиду высокой плотности упаковки гексагональные решетки широко распространены в зрительных системах различных организмов.

Выделение контуров

На рис. 6 а) приведена маска из шестиугольных элементов – «снежинка», полученная как результат последовательного применения прямой и обратной конечных разностей в трех различных направлениях осей: i , j , и k , расположенных под углом 60° друг к другу. Такая маска является дискретным аналогом трехмерной частной производной второго порядка. На рис. 6 б) приведена прямоугольная карта «снежинки», где ее оси сдвинуты до углов 45° . Разумеется, при приведенном плоском картировании неизбежны визуальные искажения, но карта позволяет проводить с ней вычисления и отображать результаты на обычном прямоугольном растре.

Другим способом отображения маски рис. 6 а) является ее представление в трехмерном пространстве в виде своеобразного «кубика Рубика», который состоит из 27 меньших кубиков – вокселей. Воксели имеют различные веса: угловые – «1», средние ребер – «-2», средние граней – «4», а центральный – «-6». При этом оси: i , j , и k , проходящие через центр «кубика Рубика» и центры боковых граней, взаимно перпендикулярны.

Кроме выражений: (1), (2) для оси – i и (4), (5) для оси – j , приведем выражения для отображения оси – k на карту обратной конечной разности:

$$p_{i,j} = e_{i,j} - e_{i-1,j-1} \quad (16)$$

и прямой конечной разности:

$$s_{i,j} = p_{i+1,j+1} - p_{i,j} \quad (17)$$

Подставляя в последнее выражение значения всех предыдущих получим аналитическое выражение для двумерной карты гексагональной маски дискретного аналога трехмерной производной второго порядка (рис. 6 б), при $Y = s$, при этом начало отсчета выберем в правом нижнем углу карты:

$$\begin{aligned} Y = & 2v_{i-3,j-3} + 2v_{i-1,j-1} + 2v_{i-2,j-3} + 2v_{i-2,j-1} + 2v_{i-3,j-2} + 2v_{i-1,j-2} - \\ & - 2v_{i-3,j-1} - 2v_{i-1,j-3} - 2v_{i-3,j-4} - 2v_{i-1,j} - 2v_{i-4,j-3} - 2v_{i,j-1} - \\ & - 6v_{i-2,j-2} + v_{i-4,j-4} + v_{i,j} + v_{i-2,j-4} + v_{i-2,j} + v_{i-4,j-2} + v_{i,j-2} \quad (18) \end{aligned}$$

На рис. 7 приведены изображения: а) тестовое – v , б) масштабированное контурное – Y , в) контрастированное контурное с обрезкой модуля значений пикселей больше 120 – Y и масштабированное и г) контурное в псевдоцветах с красными положительными и

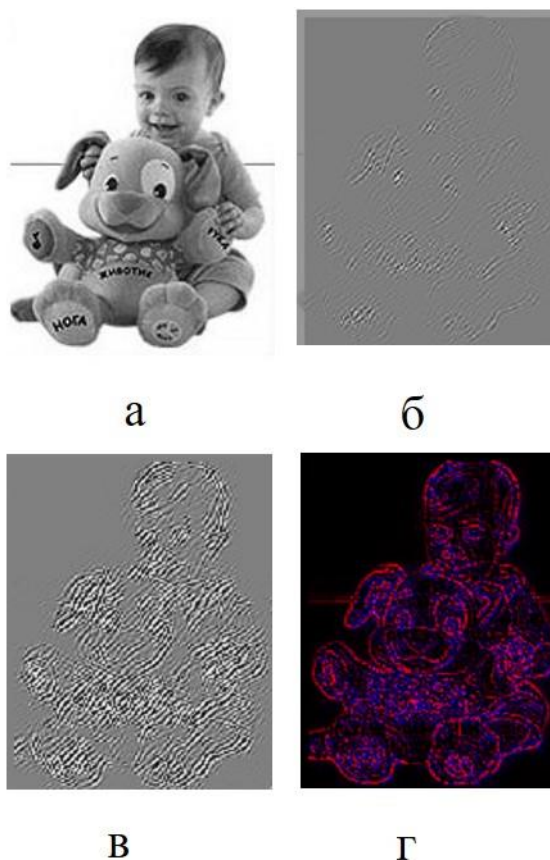


Рис.7 Изображения: а) исходное – v , б) контурное масштабированное – Y , в) искусственно контрастированное – Y , г) псевдо-цветное контурное – Y .

синими отрицательными значениями контуров – Y . Все изображения на рис. 7 получены аналогично изображениям на рис.2, но с картой гексагональной маски – «снежинки».

Сравнивая контурные (кодированные) изображения рисунков 2, 3, 4 и 7 можно видеть, что все они различны несмотря на то, что соответствуют одному исходному изображению. Их различие объясняется различием масок, с которыми производится свертка исходного изображения. Часто для оценки эффективности кодирования изображений используют оценки их информационного сжатия. В частности, *коэффициент сжатия*, представляющий собой отношение размеров выходного и входного файлов кодера или обратная ему величина *фактор сжатия*. Эти оценки достаточно трудно сделать, поскольку размер выходного файла зависит от энтропии исходного изображения, которую достаточно сложно вычислить, ввиду ее зависимости от вероятностной модели изображения, которую еще необходимо построить. С другой стороны, легко оценить интегральную мощность сигналов пикселей исходного и кодированного изображений и далее оценить *выигрыш в мощности* при кодировании взяв их отношение. Последняя оценка хотя и не связана прямо со скоростью передачи информации по каналу связи позволяет оценить снижение энергетических затрат при такой передаче. В соответствии с вышесказанным выигрыш в мощности определяется выражением:

$$W(I,C) = \frac{\sum_{j=1}^N \sum_{i=1}^M (I_{i,j})^2}{\sum_{j=1}^N \sum_{i=1}^M (C_{i,j})^2}, \quad (19)$$

где I и C – матрицы исходного и кодированного (контурного) изображения; $I_{i,j}$ и $C_{i,j}$ – элементы исходного и кодированного изображений; M и N – число строк и столбцов изображений, соответственно.

Подставляя в выражение (19) матрицы исходного изображения – v и масок E , K , L и Y , получим соответствующие выигрыши в мощности:

$$W(v,E) = 9.675 \quad W(v,K) = 2.141 \quad W(v,L) = 8.347 \quad W(v,Y) = 2.666$$

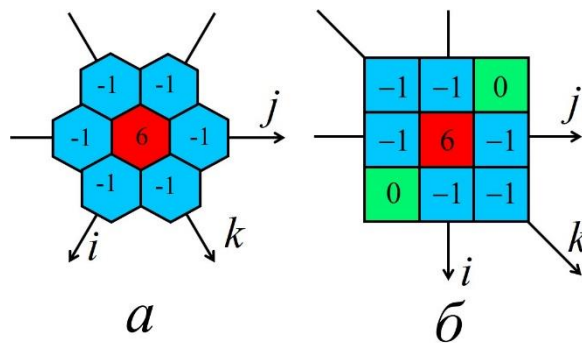


Рис.8: а) гексагональная маска «рецептивного поля» – R , б) двумерная карта «рецептивного поля» – R .

Согласно этим значениям, наибольший выигрыш достигается при использовании маски – E . Что касается маски Y , то расчет для нее производился по карте маски, а не напрямую и поэтому не подлежит сравнению с остальными масками.

Рассмотрим еще одну гексагональную маску. На рис. 8 а) приведен ее вид, а на рис. 8 б) ее прямоугольная карта. Эта маска напоминает графическое изображение рецептивного поля ганглиозной клетки сетчатки глаза и поэтому обозначим ее буквой – R . Алгоритм свертки исходного изображения – v с картой этой маски представляется выражением, которое схоже с выражением (8) за исключением двух угловых элементов, которые в карте равны нулю и веса центрального элемента:

$$R_{i,j} = 6v_{i-1,j-1} - v_{i-2,j-2} - v_{i,j} - v_{i-1,j-2} - v_{i-1,j} - v_{i-2,j-1} - v_{i,j-1} \quad (20)$$

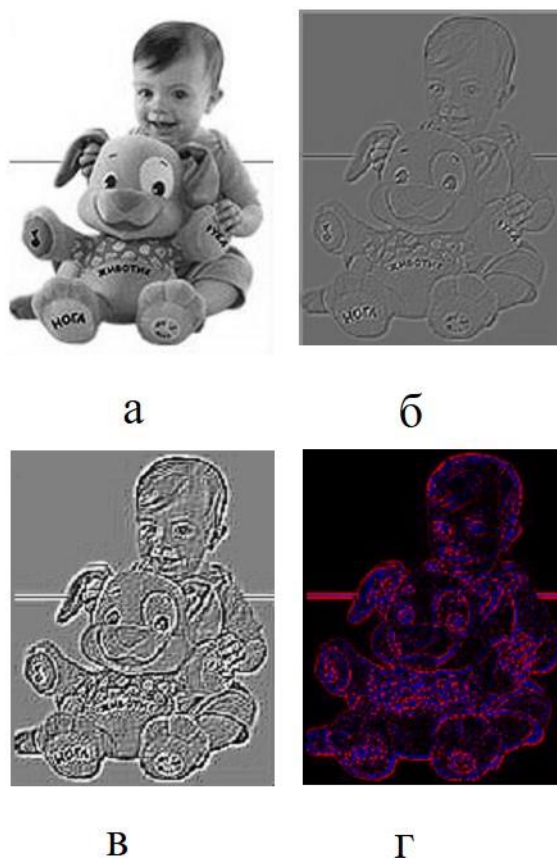


Рис.9 Изображения: а) исходное – v , б) контурное масштабированное – R , в) искусственно контрастированное – R , г) псевдо-цветное контурное – R .

Выигрыш в мощности для исходного изображения – v и маски R согласно выражению (19) равен: $W(v, R) = 3,685$, что больше, чем $W(v, Y)$. Заметим, что выигрыш в мощности для биологических систем весьма важен, поскольку дает им преимущество в выживании.

На рис. 9 приведены изображения: а) тестовое – v , б) масштабированное контурное – R , в) контрастированное контурное с обрезкой модуля значений пикселей больше 120 – R и масштабированное и г) контурное в псевдоцветах с красными положительными и синими отрицательными значениями контуров – R . Все изображения на рис. 7 получены аналогично изображениям на рис.2, но с картой гексагональной маски – «рецептивное поле» – R .

Восстановление изображения

Алгоритм восстановления (декодирования) из контурного – Y изображения исходного изображения $v = Vy$, можно получить используя уравнение (18). Из которого следует, что:

$$\begin{aligned}
 Vy_{i,j} = & Y_{i,j} - 2v_{i-1,j-1} - 2v_{i-1,j-2} - 2v_{i-2,j-1} + 2v_{i-1,j-3} + 6v_{i-2,j-2} + \\
 & + 2v_{i-3,j-1} - 2v_{i-2,j-3} - 2v_{i-3,j-2} - 2v_{i-3,j-3} + 2v_{i-3,j-4} + 2v_{i-4,j-3} + \\
 & + 2v_{i,j-1} + 2v_{i-1,j} - v_{i-2,j-4} - v_{i-4,j-2} - v_{i-4,j-4} - v_{i,j-2} - v_{i-2,j}
 \end{aligned} \tag{21}$$

При этом известны граничные значения:

$$Vy_{1,j} = Y_{1,j} = v_{1,j}, \quad Vy_{2,j} = Y_{2,j} = v_{2,j}, \quad Vy_{3,j} = Y_{3,j} = v_{3,j}, \quad Vy_{4,j} = Y_{4,j} = v_{4,j}$$

$$V_{y_{i,1}} = Y_{i,1} = v_{i,1}, V_{y_{i,2}} = Y_{i,2} = v_{i,2}, V_{y_{i,3}} = Y_{i,3} = v_{i,3}, V_{y_{i,4}} = Y_{i,4} = v_{i,4}$$

и расчет ведется для элементов исходного изображения в интервале номеров:

$$5 \leq i \leq M \text{ и } 5 \leq j \leq N.$$

Таким образом, порядок восстановления из Y – контурного изображения исходного отличается от предыдущих случаев более широкой границей, что определяется размером карты маски этого изображения.

Алгоритм восстановления (декодирования) из контурного – R изображения исходного изображения $v = Vr$, можно получить используя уравнение (20). Из которого следует, что:

$$Vr_{i,j} = 6Vr_{i-1,j-1} - Vr_{i-2,j-2} - Vr_{i-2,j-1} - Vr_{i-1,j-2} - Vr_{i,j-1} - Vr_{i-1,j} - R_{i,j} \quad (22)$$

При этом известны граничные значения:

$$Vr_{1,j} = R_{1,j} = v_{1,j}, Vr_{2,j} = R_{2,j} = v_{2,j},$$

$$Vr_{i,1} = R_{i,1} = v_{i,1}, Vr_{i,2} = R_{i,2} = v_{i,2}$$

и расчет ведется для элементов исходного изображения в интервале номеров:

$$3 \leq i \leq M \text{ и } 3 \leq j \leq N.$$

На рис. 10 показаны некоторые возможные варианты порядка восстановления небольшого исходного изображения при $M = 10$ и $N = 9$.

В варианте а) восстановление начинается с элемента, обозначенного номером «1» белым цветом, и продолжается вдоль третьей строки изображения в порядке следования обозначенных номеров элементов. Эта строка для наглядности выделена тонально (красным цветом). Следующая (четвертая) строка восстановления также выделенная тонально (зеленым цветом) начинается с номера «1», продолжается номером «2» и далее до номера «7». Последующие строки восстанавливаются в аналогичном приведенному порядку. При этом восстанавливающая маска движется вдоль строки касаясь своим нижним элементом очередного элемента изображения. Гексагональная восстанавливающая маска для первого восстановленного элемента изображения выделена тонально и обозначена буквой Ц на своем центре. Таким образом производится последовательная (поэлементная) обработка изображения. Верхняя скошенная граница данного варианта изображения имеет толщину в два элемента, боковые в один элемент, а нижняя граница отсутствует.

В варианте б) восстановление также начинается с элемента, обозначенного номером «1», затем «2», затем одновременно восстанавливаются сразу два элемента с номером «3», затем одновременно два элемента с номером «4», затем три элемента с номером «5» и далее последующие горизонтальные, тонально выделенные, строки изображения. Таким

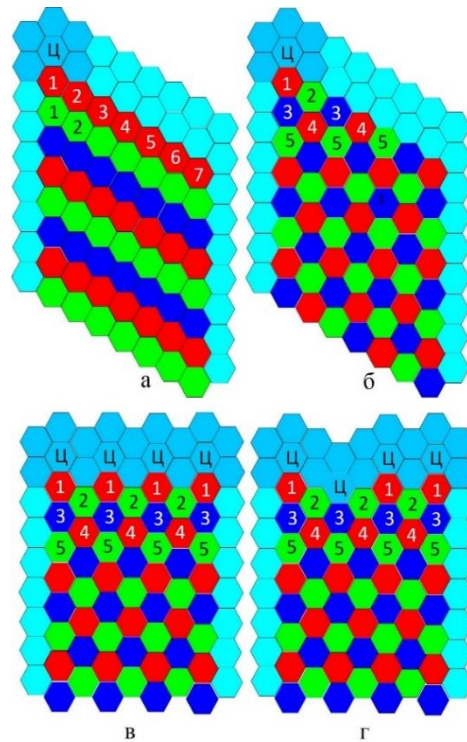


Рис.10 Гексагональная решетка.
Восстановление исходного
изображения размера 10 x 9 из
контурного того же размера: а)
последовательно, б), в) и г)
параллельно.

образом производится параллельная (построчная) обработка изображения, при движении сверху вниз «волны вычислений», фронт которой совпадает с очередной строкой изображения.

В варианте в) верхняя граница изображения уже не скошенная, а прямая. Обработка изображения ведется параллельно, построчно, при движении волны вычислений сверху вниз.

В варианте г) верхняя граница изображения не прямая, а со сдвигом вниз одного из ее столбцов. Обработка изображения, как и в двух предыдущих вариантах ведется параллельно, построчно, при движении волны вычислений сверху вниз. Этот вариант показывает, что граница изображения не обязательно должна быть прямолинейной.

Во всех вариантах важно, чтобы каждый из восстанавливаемых элементов изображения в момент восстановления входил элементом в какую-либо маску восстановления, так чтобы значения всех остальных накрываемых маской элементов изображения на момент восстановления были известны.

Отметим выигрыш по времени при параллельной обработке контурного изображения на гексагональной решетке при восстановлении, что особенно важно, когда изображения должны быстро изменяться. Например, для варианта в) число тактов времени при параллельном восстановлении равно числу восстановленных строк $M = 15$, а при последовательном, при числе элементов в строке $N = 4$, требуется $M \times N = 60$ тактов, т. е. в четыре раза больше, что снижает частоту кадров в четыре раза. При $M \approx N$ выигрыш во времени составляет величину пропорциональную квадратному корню из числа элементов изображения без границ.

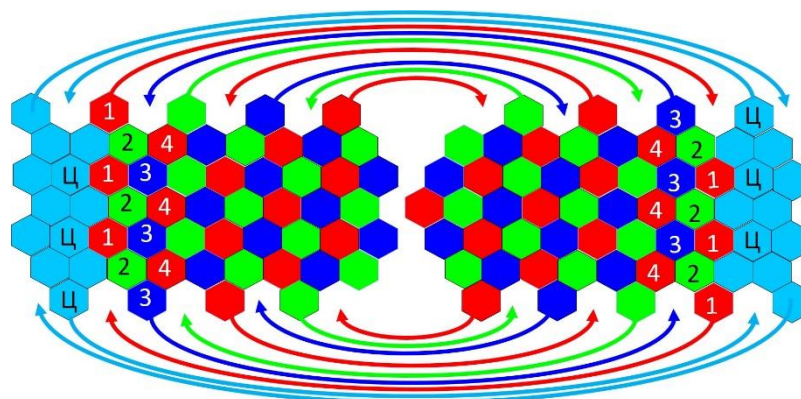


Рис.11 Упрощенная модель зрительной коры человека, состоящая из двух половин «склеенных» (в топологическом смысле) в направлениях, указанных стрелками.

Интересным примером организации решетки изображения представляется зрительная кора головного мозга человека, на которую отображается сетчатка глаза. Она разделена на две половины в левой и правой затылочных областях мозга. Каждая из двух половин зрительной коры, при ее расположении на плоскости, представляет собой слегка усеченный на одном конце эллипс [8] на который проецируется одна из двух половин сетчатки, левого или правого поля зрения. На рис.11 приведена модель такой коры. Усеченные «эллипсы» справа и слева рисунка «склеены» (в топологическом смысле) по направлениям, указанным стрелками так, что элементы в начале стрелок входят в места, указанные их остриями без перекрытия элементов. Аналогом стрелок в мозге являются нервные волокна соединяющие половины зрительной коры. После склейки образуется поверхность, напоминающая кожуру огурца с обрезанным концом. На эту поверхность топографически проецируется контурное изображение объектов, спроецированных на сетчатку хрусталиком глаза. Преобразование исходного оптического изображения на сетчатке в дискретное контурное осуществляется благодаря центрально-симметричной on-off организации рецептивных полей ганглиозных (передающих) нейронов сетчатки [4]. В описанной ситуации мы можем предположить, что контурное изображение может быть восстановлено в исходное, методом, рассмотренном в настоящей статье на ряде примеров. При этом граница изображения представляет собой замкнутое кольцо, соответствующее срезу кожуры «огурца», а кольцеобразный фронт «волны вычислений» распространяется от границы в направлении самой удаленной от границы точки поверхности коры. Заметим, что граница соответствует периферии поля зрения, которая, согласно исследованиям А. Л. Ярбуса [9], освещается равномерно – усредненно. При таком освещении достаточно передавать на границу зрительной коры только данные о средней освещенности, увеличивая таким образом *выигрыш в мощности* при контурном кодировании. Наличие волновых процессов в зрительной коре обнаружено И. А. Шевелевым [10].

Сглаживание локальных дефектов изображения

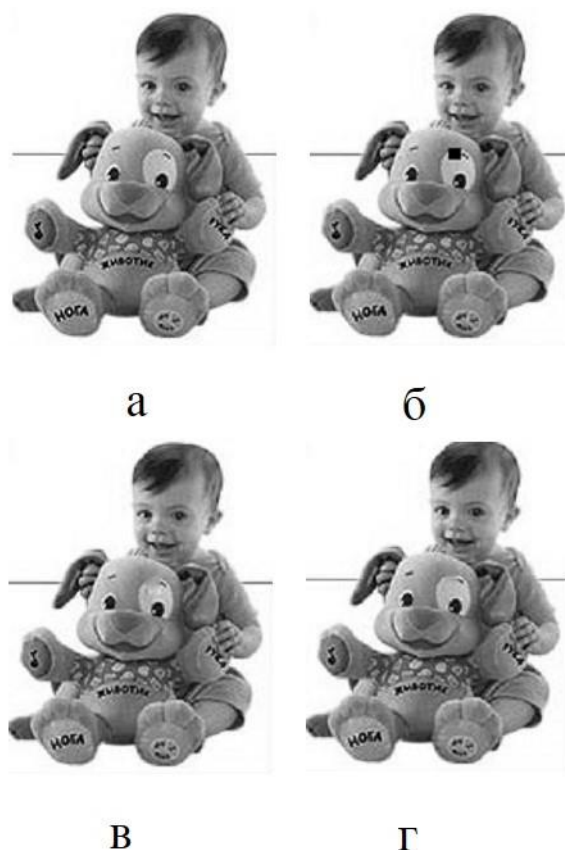


Рис.12 а) исходное изображение без дефектов, б) изображение с дефектом возле левой брови игрушки, в) сглаживание дефекта методом верхней релаксации, г) методом перекрестной интерполяции.

Изображения могут иметь локальные дефекты, в зоне которых отсутствуют либо элементы изображения (пиксели), либо их сигналы, например, при отсутствии сигналов от соответствующих элементов датчика изображения – сенселей. В системах технического зрения возможны локальные дефекты передающих матриц, а в биологических зрительных системах локальные дефекты сетчатки. Например, в сетчатке глаз человека имеются дефекты, соответствующие местам входа зрительных нервов. Воспринимая изображение окружающей среды, мы не воспринимаем эти дефекты как черные пятна, однако можем их обнаружить как «слепые пятна», при наблюдении одним глазом. В этом случае на месте этих пятен ощущается яркость окружающего их фона.

Для устранения заметности дефекта следует интерполировать на его место значение яркости (сигналов) элементов, находящихся на границе дефекта. Возможны различные алгоритмы быстрого сглаживания локальных дефектов изображения. На рис. 12 приведены изображения: а) тестовое без дефектов, б) тестовое с дефектом возле левой брови игрушки, в) со сглаженным дефектом методом *верхней релаксации* при решении задачи Дирихле – Vd, г) со сглаженным дефектом методом *перекрестной интерполяции* – Vdd.

Итерационный метод верхней релаксации используется, в частности, для численного решения уравнения Пуассона если известны значения решения на границе (*задача Дирихле*) [11]. Цифровой аналог такого уравнения приведен выше в выражении (14). Итерационное решение задачи Дирихле методом верхней релаксации для восстановления достаточно большого контурного изображения, как в приводимых здесь примерах, требует сотни итераций (прогонов). Выше был продемонстрированы прямые методы получения

исходного изображения за один прогон, а метод итераций, при небольшом числе прогонов, можно использовать для сглаживания небольшого по площади дефекта изображения.

В системе Mathcad для получения изображения – Vd со сглаженным прямоугольным дефектом размера $(k + 1) \times (q + 1)$ можно использовать программный модуль:

$$Vd := \left| \begin{array}{l} U \leftarrow v \\ \text{for } z \in 1..Q \\ \quad \text{for } j \in n..n+q \\ \quad \quad \text{for } i \in m..m+k \\ \quad \quad \quad U_{i,j} \leftarrow 0.25 \cdot \omega \cdot (U_{i-1,j} + U_{i,j-1} + U_{i+1,j} + U_{i,j+1}) + (1 - \omega) \cdot U_{i,j} \\ U \end{array} \right. \quad (23)$$

где: v – заданное перед модулем изображение с дефектом, Q – число итераций, ω – заданный весовой коэффициент, z – номер итерации, n и m – номер столбца и строки соответственно верхнего левого угла дефекта изображения, $m + k$ и $n + q$ – номера строки и столбца соответственно нижнего правого угла дефекта, U – переменная в цикле итераций. Например, изображение в) рис. 12 получено за шесть ($Q = 6$) итераций, при $\omega = 1.6$.

В итерационном методе сглаживания для вычисления значения каждого элемента сглаженного дефекта используются, по существу, все элементы исходного изображения, граничащие с дефектом. В приведенном ниже другом методе «перекрестной» интерполяции значение каждого пикселя сглаженного дефекта интерполируется по четырем элементам исходного изображения, граничащим с дефектом и лежащим на вертикали и горизонтали, проходящих через данный пиксель.

Для перекрестной интерполяции с учетом обозначений: $a = n - 1$, $b = n + q + 1$, $c = m - 1$, $d = m + k + 1$, $h = k + 2$, $l = q + 2$ и ограничений: $m \leq i \leq m + k$, $n \leq j \leq n + q$ интерполянт описывается следующим выражением:

$$\bar{v}_{i,j} = \frac{[(b-j)v_{i,a} + (j-a)v_{i,b}][i-c)(d-i) + [(d-i)v_{c,j} + (i-c)v_{d,j}][j-a)(b-j]}{l(i-c)(d-i) + h(j-a)(b-j)}. \quad (24)$$

Он используется при одновременном действии обоих указанных ограничений, а вне дефекта воспроизводится исходное изображение – $v_{i,j}$, в результате чего образуется изображение со сглаженным дефектом – Vdd.

Заключение

Рассмотрен ряд методов контурного кодирования изображений путем свертки (конволюции) их с различными центрально-симметричными масками на прямоугольной и гексагональной решетках. Все использованные маски обладают нулевым средним значением. Показано, что все полученные контурные изображения могут быть подвергнуты быстрой деконволюции с восстановлением исходных изображений если известны их граничные значения. Из чего следует, что при кодировании исходного изображения следует сохранять некоторые его границы, вид которых показан для приведенных масок кодирования. Для других масок можно выбрать сохраняемые границы руководствуясь изложенным методом выбора. Интересно отметить, что, например, в зрительной системе

человека периферия сетчатки глаз засвечивается однородным усредненным излучением, благодаря особой организации их оптической системы [12], то есть уровень граничных значений одинаков и может быть передан в зрительную кору с минимальными энергетическими затратами.

Наличие дефектов – локальных зон изображения с отсутствием сигнала изображения делает актуальной задачу их маскировки – сглаживания. Рассмотрены два метода такого сглаживания – интерполяции значений сигналов изображения в зоне дефекта с использованием значений таких сигналов от элементов изображения, граничащих с зоной дефекта. Первый из методов – *итерационной интерполяции* требует нескольких (шести в примере) циклов (кадров зоны дефекта), но обладает простотым алгоритмом итерации цикла. Второй – *перекрестной интерполяции* обеспечивает сглаживание за один цикл, но обладает более сложным алгоритмом (интерполянт), что влияет на время сглаживания, при последовательном вычислении значений элементов области дефекта.

Литература

1. Сэлмон Д. Сжатие данных, изображений и звука. М.: Техносфера, 2006.
2. Яне Б. Цифровая обработка изображений. М.: Техносфера, 2007.
3. Гонзалес Р., Вудс Р. Цифровая обработка изображений. М.: Техносфера, 2006.
4. Хьюбел Д. Глаз, мозг, зрение. – М.: Мир, 1990.
5. Марр Д. Зрение Информационный подход к изучению представления и обработки зрительных образов. – М.: Радио и связь, 1987. – 400 с.
6. Golay M. J. E. Hexagonal Pattern Transformation. *IEEE Trans. Computers*, C-18, 8, 733 – 740 (August 1969).
7. Middleton L., Sivaswamy J. *Hexagonal Image Processing: A Practical Approach*. Springer, 2005.
8. Schira M. M., Wade A. R., and Tyler C. W. Two-Dimensional Mapping of the Central and Parafoveal Visual Field to Human Visual Cortex. *J Neurophysiol* 97: 4284–4295, 2007.
9. Ярбус А. Л. Роль движений глаз в процессе зрения. – М.: Наука, 1965. – 164 с.
10. Шевелев И. А. Волновые процессы в зрительной коре мозга. *Природа*, 2001, №12.
11. Баркалов К. А. Образовательный комплекс «Параллельные численные методы». Лекционные материалы. Нижний Новгород: Нижегородский государственный университет. Факультет вычислительной математики и кибернетики, 2011.
12. Ярбус А. Л. О работе зрительной системы человека. Адекватный зрительный стимул. *Биофизика*, 20, 5, 916-920, 1975.