

УДК 004.932

МОДЕЛИ И АЛГОРИТМЫ ВНЕСЕНИЯ ДЕФОРМИРУЮЩИХ ИСКАЖЕНИЙ НА ИЗОБРАЖЕНИЯХ С ИСПОЛЬЗОВАНИЕМ РАДИАЛЬНО-БАЗИСНЫХ ФУНКЦИЙ

А. В. Акимов, М. А. Дрюченко, А. А. Сирота

Воронежский государственный университет

Поступила в редакцию 12.02.2014 г.

Аннотация. Описаны алгоритмы внесения деформирующих искажений (ДИ) в решетчатые функции дискретных аргументов с использованием различных видов интерполяции, в том числе на основе применения радиально-базисных функций (РБФ). Приведены результаты экспериментов по сравнению алгоритмов ДИ, использующих разные ее типы, а также по определению оптимальных параметров интерполяции, основанной на применении РБФ. Показана конкурентоспособность последнего способа интерполяции при внесении деформирующих искажений в цифровые изображения.

Ключевые слова: распознавание изображений, деформирующие искажения, интерполяция, радиально-базисные функции, регуляризация.

Abstract. The algorithms of imposing deformation on grid functions with discrete parameters using different kinds of interpolation including the one based on the usage of radial basis functions (RBF) are described. The results of experiments of deformation algorithms comparison using different interpolation types and experiments aimed at determining optimal parameters of interpolation based on the usage of RBF, are described. The competitiveness of the latter method of imposing deformations on digital images is demonstrated.

Keywords: image recognition, deformation, interpolation, radial basis functions, regularization.

Одной из проблем при построении алгоритмов распознавания сигналов и изображений является формирование исходных обучающих данных в виде обучающей выборки релевантных образов анализируемых объектов. Их поиск и подготовка к использованию требуют весьма значительных затрат времени. С учетом необходимости обеспечения представительности обучающих данных одним из возможных подходов является искусственное размножение обучающей выборки. В частности, для распознавания лиц на изображениях искусственное размножение эталонных об-

разов можно осуществить путем внесения в них деформирующих искажений (ДИ) стохастического характера. Данный подход может применяться во многих задачах, когда требуется провести обучение системы для распознавания объектов, меняющих свою форму с сохранением основной топологии расположения структурных элементов образа [1,2].

Математическая модель и общий алгоритм внесения ДИ. Математическая модель процесса внесения ДИ для исходной функции $f(x_1, \dots, x_n)$ n аргументов определяется следующим соотношением:

$$\begin{aligned} f(x_1 + r_1(x_1, \dots, x_n), \dots, x_n + r_n(x_1, \dots, x_n)) &= \\ = f(u_1(x_1, \dots, x_n), \dots, u_n(x_1, \dots, x_n)) &= \\ f(u(x)) = g(x_1, \dots, x_n), & \quad (1) \end{aligned}$$

© Акимов А. В., Дрюченко М. А., Сирота А. А., 2014

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 13-01-97507 р_центр_а

где $g(x_1, \dots, x_n)$ – результирующая деформированная функция; $r_i(x_1, \dots, x_n)$, $i = \overline{1, n}$ – функции вносимых деформирующих искажений по каждой координате, которые могут носить детерминированный или стохастический характер. Используя ранее введенные обозначения, а также обозначив $r(x) = (r_1(x), \dots, r_n(x))^T$, получаем:

$$f(x+r(x)) = f(u(x)) = g(x). \quad (2)$$

На рис. 1 изображен пример деформации одномерной непрерывной функции вида $f(x) = \cos(x^2/100)$ с помощью периодической функции $r(x) = 10\sin(\pi(x+13)/18)$.

При реализации процесса внесения ДИ на основе (1), (2) возникает ряд особенностей. Первая из них заключается в том, что область определения деформированной функции $g(x_1, \dots, x_n)$ не должна выходить за пределы области определения исходной функции $f(x_1, \dots, x_n)$, т.е. если для $f(x_1, \dots, x_n)$ вектор $x = (x_1, \dots, x_n)^T \in \Omega_x$, то и вектор $u(x) = (u_1(x), \dots, u_n(x))^T \in \Omega_x$. Данное условие исключает возможность возникновения артефактов при выходе аргументов в случае внесения ДИ за пределы области определения исходной функции.

Вторая особенность реализации процесса внесения ДИ на основе (1) и (2) состоит в следующем. В случаях, когда исходная функция $f(x)$ является решетчатой функцией дискретных аргументов, т.е. задана на многомерной дискретной сетке (например, в случае внесения ДИ в цифровые изображения) прямое применение (1) и (2) невозможно.

В данном случае для обеспечения возможности модификации аргументов $\hat{f}(x)$ при внесении в них ДИ произвольного характера необходимо предварительно выполнять интерполяцию и ее представление в виде функции $\tilde{f}(x)$ непрерывных аргументов. При этом общая процедура внесения ДИ будет соответствовать схеме, представленной на рис. 2. Добавим, что на последнем этапе при получении $\tilde{g}(x)$ к ошибке интерполирования добавляется погрешность дискретизации.

Утверждение. Пусть: $f(x)$ – функция, заданная на множестве значений n непрерывных аргументов Ω_x ; $\hat{f}(x)$ – решетчатая функция, сформированная на многомерной дискретной сетке, покрываемой Ω_x ; $\tilde{f}(x)$ – функция, интерполированная на основе значений $\hat{f}(x)$ и определенная на $\tilde{\Omega}_x \subset \Omega_x$; $g(x)$ – деформированная функция на основе внесения ДИ $r(x)$ в $f(x)$ по формуле (1); $\tilde{g}(x)$ – деформированная аналогичным образом функция на основе внесения ДИ $r(x)$ в $\tilde{f}(x)$; X – некое связное подмножество значений непрерывных аргументов $x \in X \subset \tilde{\Omega}_x$. Пусть также для любого $x \in X$ $u(x) \in X$.

Тогда ошибка $|g(x) - \tilde{g}(x)|$, возникающая при внесении деформации в интерполированную функцию $\tilde{f}(x)$, для $x \in X \subset \tilde{\Omega}_x$ не превысит максимальную ошибку интерполяции $|f(x) - \tilde{f}(x)|$ для $x \in X \subset \tilde{\Omega}_x$.

Доказательство. Исходя из сформулированных условий, имеем, что $g(x) = f(u(x))$ и

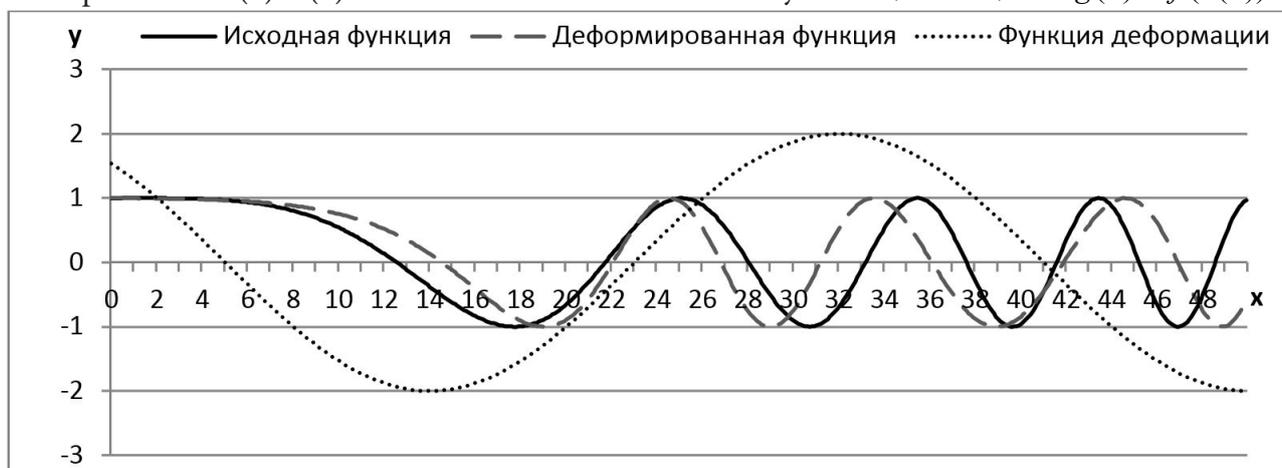


Рис. 1. Пример деформации одномерной непрерывной функции

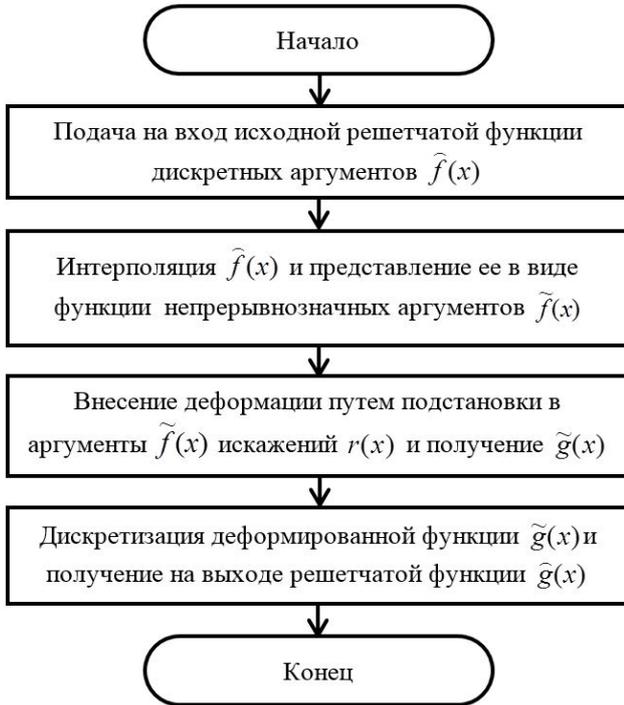


Рис. 2. Схема процесса внесения ДИ в решетчатую функцию

$\tilde{g}(x) = \tilde{f}(u(x))$. Вычтем эти два выражения друг из друга и возьмем абсолютные значения разностей в левой и правой части:

$$\Delta(x) = |g(x) - \tilde{g}(x)| = |f(u(x)) - \tilde{f}(u(x))|.$$

Пусть U – множество всех значений, принимаемых $u(x)$ внутри X , т.е. $u(x) \in U \subset X$. Таким образом, функции $g(x) = f(u(x))$ и $\Delta(x)$ определены на множестве U . При этом функция $|f(x) - \tilde{f}(x)|$ определена на множестве X . Возможно две ситуации.

Первая из них состоит в том, что функция искажения $r(x)$ может оказаться такой, что для некоторых аргументов $x^{(j)} \in X$ и $x^{(k)} \in X$ при $j \neq k$, будет выполняться $u(x^{(j)}) = u(x^{(k)})$. Действительно, достаточно, например, взять $r(x^{(j)}) = 0$, а $r(x^{(k)}) = x^{(j)} - x^{(k)}$, тогда:

$$u(x^{(j)}) = x^{(j)} + r(x^{(j)}) = x^{(j)},$$

$$u(x^{(k)}) = x^{(k)} + r(x^{(k)}) = x^{(k)} + x^{(j)} - x^{(k)} = x^{(j)}.$$

В этом случае множество $X \setminus U$ не пустое. Среди его элементов могут оказаться такие, что $|f(x) - \tilde{f}(x)|$ будет принимать свое максимальное в X значение. Тогда

$$\max |g(x) - \tilde{g}(x)| < \max |f(x) - \tilde{f}(x)|, \text{ иначе}$$

$$\max |g(x) - \tilde{g}(x)| = \max |f(x) - \tilde{f}(x)|.$$

Во втором случае если для данного искажения $r(x)$ не найдется $x^{(j)} \in X$ и $x^{(k)} \in X$ при $j \neq k$ таких, что $u(x^{(j)}) = u(x^{(k)})$, то $U = X$, так как каждому $x \in X$ соответствует свое уникальное значение $u(x) \in U$, при том, что по определению $U \subset X$, и в этом случае

$$\max |g(x) - \tilde{g}(x)| = \max |f(x) - \tilde{f}(x)|.$$

Аналогичным образом можно доказать, что

$$\min |g(x) - \tilde{g}(x)| \geq \min |f(x) - \tilde{f}(x)|.$$

Следствие. Ошибка $|g(x) - \tilde{g}(x)|$, возникающая при деформации интерполированной функции $\tilde{f}(x)$, в области ее определения $\tilde{\Omega}_x$ не выйдет за пределы максимального и минимального значения ошибки интерполяции $|f(x) - \tilde{f}(x)|$ при условии, что $u(x) \in \tilde{\Omega}_x$. Для подтверждения этого следствия достаточно повторить предыдущее доказательство в случае $X = \tilde{\Omega}_x$.

Для обеспечения выполнения условия $u(x) \in \tilde{\Omega}_x$ предлагается использовать ограничивающие значения функции $u(x)$ по краям области определения функции-маски $h(x) = (h_1(x), \dots, h_n(x))^T$:

$$u(x) = x + h(x)r(x).$$

Эти функции должны стремиться к нулю на границах области определения $\tilde{\Omega}_x$ и к единице в ее центре. В качестве них могут быть использованы гауссианы вида

$$h(x) = \frac{1}{(2\pi)^{1/2} \xi} \exp \left[-\frac{(x - \mu)^2}{2\xi^2} \right],$$

где ξ и μ – некие вещественные числа, которые всегда можно выбрать так, чтобы обеспечить выполнение указанных условий.

В дополнение к этому требуется обеспечить невозможность превышения значения величины $\max |r_i(x)|$ ($i = \overline{1, n}$) половины ширины диапазона соответствующей i компоненты области определения $\tilde{\Omega}_x$.

Линейная и кубическая интерполяция. Как уже упоминалось ранее, перед внесением ДИ в решетчатые функции дискретных аргументов требуется провести интерполяцию. Для функций одного аргумента часто используемыми методами интерполяции являются линейная и кубическая [3].

При линейной интерполяции исходный равномерно-распределенный набор точек позволяет сформировать отрезки, имеющие по две точки с каждой стороны. Далее для каждого отрезка определяются коэффициенты уравнения прямой

$$\tilde{f}(x) = ax + b, \quad (3)$$

путем подстановки известных значений для крайних точек отрезка.

Интерполируемые значения x затем подставляются в (3), в зависимости от того, внутри какого отрезка они оказываются.

Для увеличения быстродействия процесс интерполяции подразделяется на два подпроцесса: определения коэффициентов и собственно интерполяции, так как при многократных попаданиях x в один и тот же отрезок можно использовать уже рассчитанные значения коэффициентов. Этот прием может использоваться при работе с любым другим способом интерполяции. Отметим, что при деформации значение берется из того отрезка, на который выпадает рассчитанная величина $u(x)$.

В отличие от линейной при кубической интерполяции отрезки состоят из четырех точек каждый $(-1, 0, 1, 2)$. Здесь область определения исходной функции делится на отрезки так, чтобы точки 1 и 2 одного отрезка соответствовали точкам -1 и 0 следующего отрезка. Далее для каждого из них ищутся коэффициенты полинома третьего порядка:

$$\tilde{f}(x) = ax^3 + bx^2 + cx + d, \quad (4)$$

и его производной:

$$\tilde{f}'(x) = 3ax^2 + 2bx + c.$$

Для этого в полином подставляются значения исходной решетчатой функции $\hat{f}(x)$ в точках 0 и 1, а в качестве значений производных в точках 0 и 1 берутся тангенсы углов наклона прямых, проведенных между предыдущими и следующими точками. Во время самой интерполяции аналогично предыдущему линейному варианту найденные коэффициенты подставляются в (4) вместе с локальным значением x , соответствующему отрезку, в диапазон значений которого он попадает.

Для двумерных функций часто используемыми методами интерполяции являются такие, как билинейная и бикубическая ([3] и [4]). Они могут быть реализованы на основе трех проведенных последовательно друг за другом линейных или кубических интерполяций. Для интерполируемых координат (x, y) (здесь также используются локальные значения внутри отрезков по обеим осям) фиксируется значение y сначала равное 0, а затем 1 и производится интерполяция x для каждого из них. Далее полученные $\tilde{f}(x, 0)$ и $\tilde{f}(x, 1)$ используются для определения $\tilde{f}(x, y)$.

Аналогичным образом для решетчатых функций трех и более аргументов производится расширение двумерной интерполяции до трехмерной и далее.

Интерполяция с помощью радиально-базисных функций. Авторами также была исследована возможность применения радиально-базисных функций (РБФ) для решения задачи интерполяции, возникающей при деформировании решетчатых функций дискретных аргументов. Данный подход потенциально позволяет вносить непрерывные искажения $r(x)$ любого вида.

В одномерном случае при его реализации исходная решетчатая функция $\hat{f}(\hat{x}_p)$ при $p = \overline{1, N}$, подразделяется на отрезки, каждый длиной n . При этом крайние точки соседствующих друг с другом отрезков являются для них общими. Внутри отрезков используются локальные координаты. После этого для каждого из отрезков ищется аппроксимация непрерывной функцией вида

$$\tilde{f}(x) = \sum_{i=1}^n a_i \exp(-(x - \hat{x}_i)^2 / (2\sigma^2)), \quad (5)$$

где σ – параметр влияния RBF-функций; a_i – неизвестные коэффициенты. Элементам суммы соответствуют центры RBF-функций в точках \hat{x}_i (по одной функции на точку исходного дискретного набора отсчетов внутри данного отрезка).

Если для x , соответствующих позициям известных точек исходного интерполируемого отрезка, вместо $\tilde{f}(x)$ подставлять в (5) значения $\hat{f}(x)$, то для каждого отрезка полу-

чится система из n линейных уравнений (число неизвестных a_i также равно n):

$$\sum_{i=1}^n \exp(-(\hat{x}_j - \hat{x}_i)^2 / (2\sigma^2)) a_i = \hat{f}(\hat{x}_j), \quad j = \overline{1, n},$$
 или в матричном виде: $EA = F$, где матрица E состоит из элементов $\exp(-(\hat{x}_j - \hat{x}_i)^2 / (2\sigma^2))$ со столбцами, пронумерованными $i = \overline{1, n}$, и строками – $j = \overline{1, n}$, вектор $A = (a_1, \dots, a_n)^T$, а вектор $F = (\hat{f}(\hat{x}_1), \dots, \hat{f}(\hat{x}_n))^T$.

Длина отрезка и параметр влияния RBF-функций σ подбираются экспериментальным путем. Получаемые таким образом системы уравнений не всегда могут иметь стандартное решение $A = E^{-1}F$ из-за плохой обусловленности матрицы E . В этих случаях применяют методы регуляризации. Например, используют расчет псевдообратной матрицы $A = E^+ F$ (где E^+ – псевдообратная матрица к E) или регуляризацию по А. Н. Тихонову [5]. В последнем случае решение имеет вид

$$A = (\lambda I + E^T E)^{-1} E^T (F - EA_0),$$

где I – единичная матрица размером $n \times n$; λ – параметр регуляризации, A_0 – априорное решение. При регуляризации, чем больше λ , тем лучше обусловленность и ближе решение к априорной оценке A_0 , но дальше от решения исходной некорректной задачи. Исходя из этих соображений, параметр λ должен принимать промежуточные значения, определяемые эмпирическим путем или при помощи некоторого анализа. Если в качестве A_0 использовать нулевую матрицу, то регуляризация будет проводиться без учета априорного решения.

Для функций двух аргументов процесс во многом аналогичен одномерному случаю. Исходная матрица значений решетчатой функции $\hat{f}(\hat{x}_p, \hat{y}_q)$ при $p = \overline{1, M}$ и $q = \overline{1, N}$ подразделяется на блоки размерами $m \times n$. (При этом крайние точки соседствующих друг с другом блоков являются для них общими, и используются локальные координаты вдоль каждой из осей). После чего для каждого из блоков ищется аппроксимация непрерывной функцией вида

$$\hat{f}(x, y) = \sum_{i=1}^n \sum_{j=1}^m a_{i,j} \exp(-((x - \hat{x}_i)^2 + (y - \hat{y}_j)^2) / (2\sigma^2)). \quad (6)$$

Элементам суммы соответствуют центры RBF-функций с координатами \hat{x}_i и \hat{y}_j (по одной функции на точку блока исходной матрицы). Если для x и y , соответствующих координатам точек в блоке, вместо $\hat{f}(x, y)$ подставлять в (6) соответствующие этим точкам значения $\hat{f}(x, y)$, то для каждого блока получится система из $m \times n$ линейных уравнений (число неизвестных также равно $m \times n$):

$$\sum_{i=1}^n \sum_{j=1}^m \exp(-((\hat{x}_k - \hat{x}_i)^2 + (\hat{y}_l - \hat{y}_j)^2) / (2\sigma^2)) a_{i,j} = \hat{f}(\hat{x}_k, \hat{y}_l), \\ k = \overline{1, m}, \quad l = \overline{1, n}.$$

В матричном виде получим уравнение

$$EA = F,$$

где матрица E состоит из элементов $\exp(-((\hat{x}_k - \hat{x}_i)^2 + (\hat{y}_l - \hat{y}_j)^2) / (2\sigma^2))$ со столбцами, пронумерованными комбинациями значений индексов $i = \overline{1, m}$ и $j = \overline{1, n}$, и строками, пронумерованными комбинациями $k = \overline{1, m}$ и $l = \overline{1, n}$; вектор $A = (a_{11}, \dots, a_{mn})^T$; вектор $F = (\hat{f}(\hat{x}_1, \hat{y}_1), \dots, \hat{f}(\hat{x}_m, \hat{y}_n))^T$ содержит значения исходной решетчатой функции, пронумерованные путем комбинирования значений индексов для каждой из осей матрицы. Аналогичным образом можно выполнить интерполяцию решетчатых функций с любым числом аргументов.

Результаты экспериментов. Выбор длины отрезка n для одномерного случая, размеров блоков $m \times n$ для двумерного случая, параметра влияния RBF-функции σ и параметра регуляризации λ (в случае ее использования) осуществлялся путем оценки невязки интерполированной функции $\tilde{f}(x)$ и исходной функции $f(x) - |f(x) - \tilde{f}(x)|$ для разных значений перечисленных параметров. С этой целью авторы использовали непрерывные функции $f(x)$, проводили их дискретизацию и по полученным результатам строили интерполяции $\tilde{f}(x)$. На рис. 3 изображена зависимость усредненной невязки (в данном случае средней модульной ошибки интерполирования) от λ при фиксированных $n = 2$ и $\sigma = 2$ для функции $f(x) = \cos(x^2 / w)$ в области от 1 до 100 с шагом интерполирования 1 и шагом проверки величиной в 0.1 при постепенном изменении w от 1 до 300.

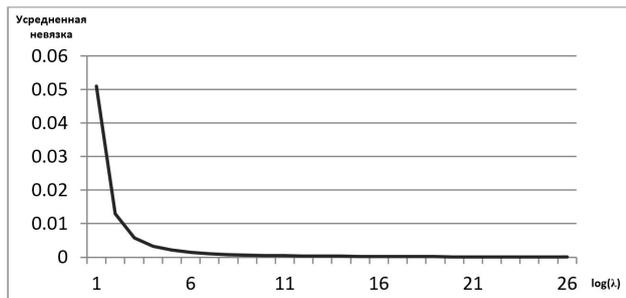


Рис. 3. Зависимость усредненной невязки от λ при фиксированных $n = 2$ и $\sigma = 2$ для функции $f(x) = \cos(x^2 / w)$ в области от 1 до 100 с шагом интерполирования 1 и шагом проверки величиной в 0.1 при постепенном изменении w от 1 до 300

В ходе тестирования было выявлено, что для функций одного аргумента хорошего приближения можно достичь, используя размер блока $n = 25$ и параметр влияния RBF-функций $\sigma = 1$. Эксперименты показали, что псевдообращение и регуляризация (при $\lambda = 10^{-16}$) обеспечивают примерно одну и ту же точность интерполяции, но при этом использование последней занимает меньше времени. Также было выяснено, что на общих границах отрезков усреднение значений возвращаемых каждым из них для этих точек оказывает ничтожно малое влияние на общий результат.

При проведении экспериментов по оценке производительности алгоритмов в качестве испытательного стенда персональный компьютер со следующими характеристиками: процессор Intel Core i5-3210M 2.5 ГГц с возможностью ускорения во время выполнения до 3.1 ГГц; оперативная память 6 ГБ DDR3; ОС – 64-х битная версия Windows 8.1; среда разработки Matlab R2013b.

Тестирование разных способов интерполяции показало, что для разных типов и частот функций тот или иной вариант интерпо-

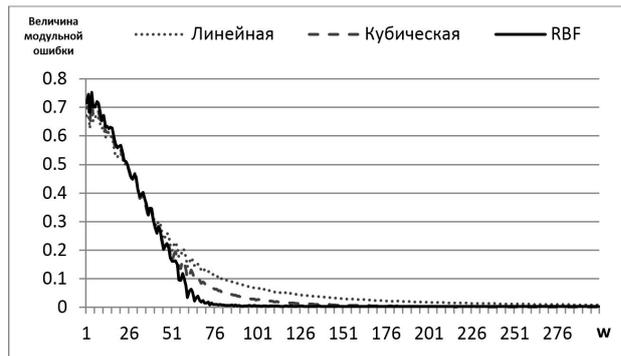


Рис. 4. Сравнение точности интерполяции для функции $f(x) = \cos(x^2 / w)$ в области от 1 до 100 с шагом интерполирования 1 и шагом проверки величиной в 0.1 при постепенном изменении w от 1 до 300

ляции выдает разные значения ошибок. Так сравнение точности интерполяции для функции $f(x) = \cos(x^2 / w)$ в области от 1 до 100 с шагом интерполирования 1 и шагом проверки величиной в 0.1 при постепенном изменении w от 1 до 300 показывает зависимости, представленные на рис. 4.

Также были произведены замеры скорости работы алгоритмов деформации, использующих неодинаковые способы интерполяции, при внесении ДИ в функцию $f(x) = \cos(x^2 / w)$ (табл. 1). Тестирование производилось для разных границ области определения.

Для двумерного случая в ходе тестирования было выявлено, что достаточно хорошего приближения можно достичь, если использовать параметр влияния RBF-функций $\sigma = 2$, размер блока 4×4 и проводить расчет псевдообратной матрицы для решения получаемых систем линейных уравнений. Эксперименты также показали, что при «сшивке» на границах блоков применение усреднения значений полученной интерполяции не улучшает результат. На рис. 5 показаны примеры ре-

Таблица 1

Сравнение быстродействия различных способов интерполяции, с

Тип интерполяции / Границы области определения	Билинейная	Бикубическая	RBF (расчет псевдообратной матрицы)	RBF (регуляризация по Тихонову)
От 1 до 50	0.0018	0.0024	0.0039	0.0033
От 1 до 100	0.0033	0.0044	0.007	0.0059
От 1 до 500	0.0154	0.0211	0.0328	0.0278

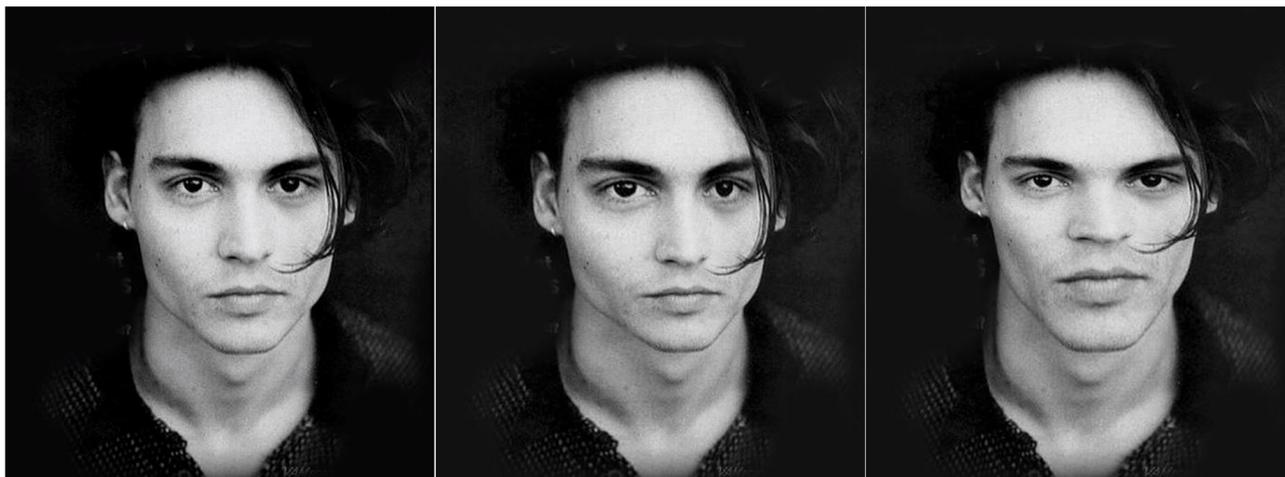


Рис. 5. Примеры результатов внесения ДИ с применением основанной на RBF-функциях интерполяции и периодической функции искажения (первое изображение является исходным)

зультатов внесения ДИ в цифровое изображение с использованием интерполяции на основе RBF-функций и периодической функции искажения.

Было произведено сравнение быстродействия различных вариантов внесения деформации в изображения (табл. 2), основанных на разных способах интерполяции.

Также была осуществлена оценка точности разных способов интерполяции (табл. 3). Для этого они тестировались в приложении к задаче изменения размера изображения. При этом производилось повторное увеличение его предварительно уменьшенной копии (32×24 пикселя) и сравнение полученного результата с оригиналом (320×240 пикселей) для каждого из способов.

Аналогично одномерному случаю здесь также в зависимости от вида и частоты исходной функции по степени приближения лидируют разные способы интерполяции. Так для $f(x, y) = \cos((x^2 + y^2) / w)$ в области от 1 до 50 по обеим координатам с шагом интерполирования 1 и шагом проверки величиной в 0.1 при постепенном изменении w от 1 до 200 получают зависимости, представленные на рис. 6.

Исходя из представленных результатов, можно сделать вывод, что использование предложенного подхода внесения деформации в изображения на основе радиально-базисных функций показывает сравнимые результаты с другими методами интерполяции, а ряде случаев превосходит их по точности, и является

Таблица 2

Сравнение быстродействия различных способов интерполяции при деформации реальных изображений, с

Тип интерполяции / Размер черно-белого изображения (в пикселях)	Билинейная	Бикубическая	RBF
160×120	0,85	1,6	1,9
320×240	9,22	17,67	20,92
640×480	27,44	54	60,2

Таблица 3

Сравнение точности разных способов интерполяции в приложении к задаче изменения размера изображения

Тип интерполяции	Билинейная	Бикубическая	RBF
Величина среднемодульной ошибки для 8-ми битного кодирования черно-белого изображения	14,64	13,9	14,08

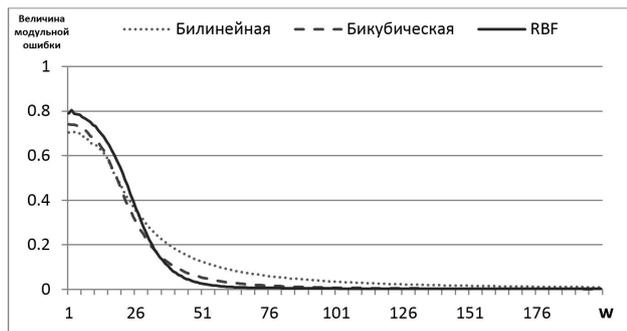


Рис. 6. Сравнение точности интерполяции для функции $f(x, y) = \cos((x^2 + y^2) / w)$ в области от 1 до 50 по обеим координатам с шагом интерполяции 1 и шагом проверки величиной в 0.1 при постепенном изменении w от 1 до 200 перспективным способом формирования обучающих выборок в ходе решения различных задач распознавания изображений.

СПИСОК ЛИТЕРАТУРЫ

1. Запрягаев С. А. Распознавание рукописных символов на основе анализа дескрипторов функций длины хорды / С. А. Запрягаев, А. И. Сорокин // Вестник Воронеж. гос. ун-та, Сер.: Системный анализ и информационные технологии. – 2009. – № 2. – С. 49–58.

Акимов А. В. – магистр технических наук, аспирант кафедры Технологий обработки и защиты информации, факультет компьютерных наук, Воронежский государственный университет. E-mail: frinatella@gmail.ru

Дрюченко М. А. – к.т.н., ассистент кафедры Технологий обработки и защиты информации, факультет компьютерных наук, Воронежский государственный университет. E-mail: m_dryuchenko@mail.ru

Сирота А. А. – д.т.н., профессор, заведующий кафедрой Технологий обработки и защиты информации, факультет компьютерных наук, Воронежский государственный университет. E-mail: sir@cs.vsu.ru

2. Сирота А. А. Модели и алгоритмы классификации многомерных данных на основе нейронных сетей с радиально-базисными функциями / А. А. Сирота, А. В. Цуриков // Вестник Воронеж. гос. ун-та, Сер.: Системный анализ и информационные технологии. – 2013. – № 1. – С. 154–161.

3. Бикубическая интерполяция, теория и практическая реализация. – <http://habrahabr.ru/post/111402/>. – (08.01.2014 г.).

4. Гонсалес Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс; пер. с англ. П. А. Чочиа. – М. : Техносфера, 2005. – 1072 с.

5. Сизиков В. С. Устойчивые методы обработки результатов измерений. Учебное пособие / В. С. Сизиков. – СПб. : СпецЛит, 1999. – 240 с.

Akimov A. V. – Magister of Technical Sciences, Aspirant, Department of Processing Technology and Information Security, Computer Sciences Faculty, Voronezh State University. E-mail: frinatella@gmail.ru

Dryuchenko M. A. – Candidate of Technical Sciences, Assistant Professor, Department of Processing Technology and Information Security, Computer Sciences Faculty, Voronezh State University. E-mail: m_dryuchenko@mail.ru

Sirota A. A. – Doctor of Technical Sciences, Professor, Department of Processing Technology and Information Security, Computer Sciences Faculty, Voronezh State University. E mail: sir@cs.vsu.ru