УДК 517.514

В. С. МУХА, А. Н. КУЗЬКОВ

ЛИНЕЙНОЕ ПРОПОРЦИОНАЛЬНОЕ ПРОКРУСТОВО ПРЕОБРАЗОВАНИЕ В РАСПОЗНАВАНИИ РУКОПИСНЫХ БУКВ

Белорусский государственный университет информатики и радиоэлектроники, Минск, Беларусь,
e-mail: mukha@bsuir.by

Разработано линейное пропорциональное преобразование обычных (двумерных) матриц. Преобразование реализовано в виде компьютерной программы и применено для распознавания рукописных букв кириллицы. Предложенное преобразование показало более высокую достоверность распознавания в сравнении с другими преобразованиями.

Ключевые слова: Прокрустово преобразование, распознавание рукописных букв, многомерная линейная регрессия, многомерные матрицы.

V. S. MUKHA, A. N. KUZKOU

LINEAR PROPORTIONAL PROCRUSTES TRANSFORMATION IN RECOGNITION OF HANDWRITTEN LETTERS

Belarusian State University of Informatics and Radioelectronics, Minsk, Belarus,
e-mail: mukha@bsuir.by

The linear proportional transformation of two-dimensional matrices was developed. This transformation was implemented as a computer program and was applied to recognize handwritten Cyrillic letters. The proposed transformation showed higher recognition accuracy in comparison with other transformations.

Keywords: Procrustes transformation, recognition of handwritten letters, multi-variate linear regression, multi-dimensional matrix.

Введение. Прокрустово преобразование представляет интерес для решения различных задач распознавания графических представлений объектов, в частности печатных и рукописных букв и цифр. В связи со сложностью программной реализации известного ортогонального Прокрустова преобразования [1] возникает необходимость в поиске и программной реализации альтернативных преобразований для решения задач различных классов. Например, ортогональное Прокрустово преобразование успешно может быть заменено линейным Прокрустовым преобразованием в задаче распознавания печатных букв [2–4]. Однако при распознавании рукописных цифр обнаружены определенные недостатки линейного Прокрустова преобразования, состоящие в неограниченных возможностях масштабирования по каждой координате изображения [5]. Для преодоления этих проблем в [5] предложено пропорциональное Прокрустово преобразование. В настоящей работе дается теоретическое обоснование пропорционального Прокрустова преобразования и выполняется более детальное его исследование при распознавании рукописных букв.

1. Линейное пропорциональное преобразование случайных векторов. Дадим сначала постановку и решение задачи линейного пропорционального преобразования двух случайных векторов. Пусть $\mathring{x} = (\mathring{x}_1, \mathring{x}_2, ..., \mathring{x}_k)$, $\mathring{y} = (\mathring{y}_1, \mathring{y}_2, ..., \mathring{y}_k)$ – центрированные случайные векторы (одномерные центрированные случайные матрицы k-го порядка [6]), и

$$\mathring{z} = {}^{0,1}(\beta \mathring{x}) \quad (1)$$

– линейное преобразование вектора $\mathring{x}$ в вектор $\mathring{z}$ с матрицей преобразования $\beta = (\beta_{i,j})$, $i, j = \overline{1,k}$. Здесь и ниже мы будем использовать многомерно-матричный математический аппарат работы [6], так что ${}^{0,1}(\beta \overset{\circ}{x})$ означает (0,1)-свернутое произведение двумерной матрицы β и одномерной матрицы $\overset{\circ}{x}$. Сформулируем следующую оптимизационную задачу:

$$f(\beta) = \mathrm{tr} D \to \min_{\beta}, \tag{2}$$

$$\psi_m(\beta) = \beta_{m,m} - \beta_{m+1,m+1} = 0 \ \ \forall m = \overline{1, k-1}, \tag{3}$$

$$D = E\left({}^{0,0}\left(\left(\overset{\circ}{y} - {}^{0,1}(\beta \overset{\circ}{x}) \right)\left(\overset{\circ}{y} - {}^{0,1}(\overset{\circ}{x} \beta^T) \right) \right) \right), \tag{4}$$

$\mathrm{tr}D$ – след матрицы D, $T = \begin{pmatrix} i, j \\ j, i \end{pmatrix}$ – подстановка транспонирования двумерной матрицы β [6].

Будем считать, что случайный вектор z как результат преобразования (1) имеет то же среднее значение, что и случайный вектор y. Тогда искомое преобразование для произвольных (не центрированных) векторов x и y будет иметь вид

$$z = \nu_y + {}^{0,1}(\beta(x - \nu_x)), \tag{5}$$

где $\nu_x = E(x)$, $\nu_y = E(y)$ – средние значения векторов x и y соответственно.

Т е о р е м а 1. *Элементы* $\beta_{1,1}$ *и* $\beta_{i,j}$, $i, j = \overline{1,k}$, $i \neq j$, *матрицы* β *в* (1), *обеспечивающие решение задачи* (2)–(4), *определяются как решение системы* $k^2 - k + 1$ *линейных алгебраических уравнений* (*СЛАУ*) *вида*

$$\begin{cases} \beta_{1,1} \sum\limits_{m=1}^{k} \mu_{x_m x_m} + \sum\limits_{m=1}^{k} \sum\limits_{\substack{l=1,\\ l\neq m}}^{k} \beta_{m,l} \mu_{x_l x_m} = \sum\limits_{m=1}^{k} \mu_{y_m x_m}, \ (i = j = 1), \\ \beta_{1,1} \mu_{x_i x_j} + \sum\limits_{\substack{1,\\ l \neq i}} \beta_{i,l} \mu_{x_l x_j} = \mu_{y_i x_j}, \ i, j = \overline{1,k}, \ i \neq j, \end{cases} \tag{6}$$

где

$$\mu_{yx} = (\mu_{y_i x_j}) = \left(E(\overset{\circ}{y}_i \overset{\circ}{x}_j) \right), \ \mu_{x^2} = (\mu_{x_i x_j}) = \left(E(\overset{\circ}{x}_i \overset{\circ}{x}_j) \right), \ i, j = \overline{1,k}. \tag{7}$$

Д л я д о к а з а т е л ь с т в а теоремы введем функцию Лагранжа

$$F(\beta^T, \lambda_1, \ldots, \lambda_{k-1}) = \mathrm{tr}\, D + \sum_{m=1}^{k-1} \lambda_m (\beta_{m,m} - \beta_{m+1,m+1}),$$

где $\lambda_1, \lambda_2, \ldots, \lambda_{k-1}$ – неопределенные множители Лагранжа. В соответствии с методом множителей Лагранжа решение задачи (2), (3) сводится к системе уравнений:

$$\begin{cases} \dfrac{dF(\beta^T, \lambda_1, \ldots, \lambda_{k-1})}{d\beta^T} = \dfrac{d(\mathrm{tr}D)}{d\beta^T} + \dfrac{d\left(\sum\limits_{m=1}^{k-1} \lambda_m (\beta_{m,m} - \beta_{m+1,m+1}) \right)}{d\beta^T} = 0, \\ \dfrac{dF(\beta^T, \lambda_1, \ldots, \lambda_{k-1})}{d\lambda_l} = \dfrac{d\left(\sum\limits_{m=1}^{k-1} \lambda_m (\beta_{m,m} - \beta_{m+1,m+1}) \right)}{d\lambda_l} = \beta_{l,l} - \beta_{l+1,l+1} = 0, \ \ l = \overline{1, k-1}, \end{cases}$$

где дифференцирование выполняется в соответствии с теорией многомерно-матричного дифференцирования работы [6]. Для матрицы D (4) имеем выражение

$$D = E\left({}^{0,0}\left(\left(\overset{\circ}{y} - {}^{0,1}(\beta \overset{\circ}{x}) \right)\left(\overset{\circ}{y} - {}^{0,1}(\overset{\circ}{x} \beta^T) \right) \right) \right) = \mu_{y^2} - 2\, {}^{0,1}\left(\mu_{yx} \beta^T \right) + {}^{0,1}\left(\beta\, {}^{0,1}\left(\mu_{x^2} \beta^T \right) \right),$$

где $\mu_{x^2} = (\mu_{x_i x_j}) = E({}^{0,0}(\mathring{x})^2)$, $\mu_{y^2} = (\mu_{y_i y_j}) = E({}^{0,0}(\mathring{y})^2)$, $\mu_{yx} = (\mu_{y_i x_j}) = E({}^{0,0}(\mathring{y}\mathring{x}))$ – моменты векторов $\mathring{x}$ и $\mathring{y}$; ${}^{0,0}(\mathring{x})^2 = {}^{0,0}(\mathring{x}\mathring{x})$, ${}^{0,0}(\mathring{y})^2 = {}^{0,0}(\mathring{y}\mathring{y})$ – $(0,0)$-свернутые квадраты матриц; ${}^{0,0}(\mathring{y}\mathring{x})$ – $(0,0)$-свернутое произведение матриц.

Найдем сначала производную $d(\mathrm{tr}D)/d\beta^T$. В соответствии с теоремой о производной следа многомерной матрицы (см. приложение), имеем

$$\frac{d(\mathrm{tr}D)}{d\beta^T} = {}^{0,2}\left(\frac{d\mathrm{tr}\,D}{dD}\frac{dD}{d\beta^T}\right) = {}^{0,2}\left(E(0,1)\frac{dD}{d\beta^T}\right), \tag{8}$$

где ${}^{0,2}\left(E(0,1)\dfrac{dD}{d\beta^T}\right)$ – $(0,2)$-свернутое произведение матриц, $E(0,1)$ – $(0,1)$ – единичная матрица [6]. Будем искать теперь производную $dD/d\beta^T$ для D вида (4). Для этого найдем вспомогательные производные. Учитывая, что $d\beta^T / d\beta^T = E(0,2)$, $E(0,2)$ – $(0,2)$-единичная матрица [6], получим

$$\frac{d}{d\beta^T}\,{}^{0,1}\left(\mu_{yx}\beta^T\right) = {}^{0,1}\left(\mu_{yx}E(0,2)\right),$$

$$\frac{d}{d\beta^T}\,{}^{0,1}\left(\mu_{x^2}\beta^T\right) = {}^{0,1}\left(\mu_{x^2}E(0,2)\right),$$

$$\frac{d}{d\beta^T}\,{}^{0,1}\left(\beta\,{}^{0,1}\left(\mu_{x^2}\beta^T\right)\right) = {}^{0,1}\left(\left(\frac{d\beta}{d\beta^T}\right)^{B_{4,2}}\,{}^{0,1}\left(\mu_{x^2}\beta^T\right)\right)^{B_{4,2}} + {}^{0,1}\left(\beta\left(\frac{d}{d\beta^T}\,{}^{0,1}\left(\mu_{x^2}\beta^T\right)\right)\right),$$

где $B_{4,2}$ – подстановка транспонирования на четырех индексах типа «вперед» [6], которая, в соответствии с ее определением, имеет вид $B_{4,2} = \begin{pmatrix} i,j,k,l \\ k,l,i,j \end{pmatrix}$. Учитывая, что $\dfrac{d\beta}{d\beta^T} = \dfrac{d\left(\beta^T\right)^T}{d\beta^T} = \left(\dfrac{d\beta^T}{d\beta^T}\right)^{(T,E_2)} = E(0,2)^{(T,E_2)}$, где $E_2 = \begin{pmatrix} i,j \\ i,j \end{pmatrix}$ – тождественная подстановка на двух индексах, $(T,E_2) = \begin{pmatrix} i,j,k,l \\ j,i,k,l \end{pmatrix}$ – составная подстановка, будем иметь

$$\left(\frac{d\beta}{d\beta^T}\right)^{B_{4,2}} = E(0,2)^{(T,E_2)*B_{4,2}} = E(0,2)^{T_1},$$

где $T_1 = (T,E_2) * B_{4,2}$ – суперпозиция подстановок (T,E_2) и $B_{4,2}$, имеющая вид $T_1 = \begin{pmatrix} i,j,k,l \\ l,k,i,j \end{pmatrix}$. Тогда

$$\frac{d}{d\beta^T}\,{}^{0,1}\left(\beta\,{}^{0,1}\left(\mu_{x^2}\beta^T\right)\right) = {}^{0,1}\left(E(0,2)^{T_1}\,{}^{0,1}\left(\mu_{x^2}\beta^T\right)\right)^{H_{4,2}} + {}^{0,1}\left(\beta\,{}^{0,1}\left(\mu_{x^2}E(0,2)\right)\right).$$

На основании полученных вспомогательных результатов будем иметь

$$\frac{dD}{d\beta^T} = -2\,{}^{0,1}\left(\mu_{yx}E(0,2)\right) + {}^{0,1}\left(E(0,2)^{T_1}\,{}^{0,1}\left(\mu_{x^2}\beta^T\right)\right)^{H_{4,2}} + {}^{0,1}\left(\beta\,{}^{0,1}\left(\mu_{x^2}E(0,2)\right)\right) = (z_{i,j,k,l}).$$

Можно показать, что с учетом симметричности матрицы D второе слагаемое данного равенства равно третьему, так что производная матрицы D приобретает вид

$$\frac{dD}{d\beta^T} = -2\,{}^{0,1}\left(\mu_{yx}E(0,2)\right) + 2\,{}^{0,1}\left(\beta\,{}^{0,1}\left(\mu_{x^2}E(0,2)\right)\right).$$

С учетом выражения (8) будем иметь следующую производную следа матрицы D:

$$\frac{d(\operatorname{tr} D)}{d\beta^T} = {}^{0,2}\!\left(E(0,1)\frac{dD}{d\beta^T} \right) = -2\mu_{yx} + 2\,{}^{0,1}\!\left(\beta\mu_{x^2}\right).$$

Найдем также производную $d\left(\sum_{m=1}^{k-1}\lambda_m\left(\beta_{m,m}-\beta_{m+1,m+1}\right)\right)/\,d\beta^T$. Поскольку

$$\frac{d\beta_{m,m}}{d\beta^T} = A_m = (a_{i,j}) = \left(\begin{cases}1, & \text{если } i=j=m,\\ 0, & \text{иначе}\end{cases}\right), i,j=\overline{1,k},\ m=\overline{1,k-1},$$

$$\frac{d\beta_{m+1,m+1}}{d\beta^T} = B_m = (b_{i,j}) = \left(\begin{cases}1, & \text{если } i=j=m+1,\\ 0, & \text{иначе}\end{cases}\right), i,j=\overline{1,k},\ m=\overline{1,k-1},$$

то

$$\frac{d\left(\lambda_m(\beta_{m,m}-\beta_{m+1,m+1})\right)}{d\beta^T} = \lambda_m(A_m-B_m) = C_m,$$

где

$$C_m = (c_{i,j}) = \begin{pmatrix} \lambda_m, & \text{если } i=j=m,\\ -\lambda_m, & \text{если } i=j=m+1,\\ 0, & \text{иначе}\end{pmatrix}, m=\overline{1,k-1}.$$

Таким образом,

$$\frac{d\left(\sum_{m=1}^{k-1}\lambda_m(\beta_{m,m}-\beta_{m+1,m+1})\right)}{d\beta^T} = \sum_{m=1}^{k-1} C_m = G,\ m=\overline{1,k-1}. \tag{9}$$

В развернутом виде матрица G (9) имеет следующий вид:

$$G=(g_{i,j})=\begin{pmatrix}\lambda_1 & 0 & 0 & 0 & 0 & 0\\ 0 & -\lambda_1+\lambda_2 & 0 & 0 & 0 & 0\\ 0 & 0 & -\lambda_2+\lambda_3 & 0 & 0 & 0\\ \dots & \dots\dots & \dots\dots & \dots & \dots\dots & \dots\\ 0 & 0 & 0 & 0 & -\lambda_{k-2}+\lambda_{k-1} & 0\\ 0 & 0 & 0 & 0 & 0 & -\lambda_{k-1}\end{pmatrix}, i,j=\overline{1,k}.$$

Элементы этой матрицы определяются формулой

$$g_{i,j}=\begin{cases}\lambda_1, \text{ если } i=j=1,\\ \lambda_i-\lambda_{i-1}, \text{ если } i=j=\overline{2,k-1},\\ -\lambda_{k-1}, \text{ если } i=j=k,\\ 0, \quad \text{иначе.}\end{cases}$$

Таким образом, для решения поставленной задачи (2), (3) нам необходимо решить систему уравнений

$$\begin{cases}-2\mu_{yx}+2\,{}^{0,1}\!\left(\beta\mu_{x^2}\right)+G=0,\\ \beta_{m,m}-\beta_{m+1,m+1}=0, \quad m=\overline{1,k-1},\end{cases} \tag{10}$$

относительно элементов матриц β и G. Упростим данную систему уравнений. Для этого в уравнениях первого (матричного) уравнения системы (10) положим $\beta_{i,i}=\beta_{1,1}\ \forall i=\overline{1,k}$. Это приведет к исключению второго (скалярного) уравнения системы (10). С учетом обозначений (7) второе слагаемое в левой части первого уравнения системы (10) представляется в виде

$$z = {}^{0,1}\left(\beta\mu_{x^2}\right) = (z_{i,j}) = \left(\sum_{l=1}^{k}\beta_{i,l}\mu_{x_l x_j}\right) = \left(\beta_{1,1}\mu_{x_i x_j} + \sum_{l=1}^{i-1}\beta_{i,l}\mu_{x_l x_j} + \sum_{l=i+1}^{k}\beta_{i,l}\mu_{x_l x_j}\right).$$

В результате вместо системы уравнений (10) получим следующую систему уравнений:

$$-2\mu_{y_i x_j}\mu_{i,j} + 2\beta_{1,1}\mu_{x_i x_j} + 2\sum_{l=1}^{i-1}\beta_{i,l}\mu_{x_l x_j} + 2\sum_{l=i+1}^{k}\beta_{i,l}\mu_{x_l x_j} + g_{i,j} = 0,\ i,j = \overline{1,k}. \tag{11}$$

В развернутом виде система уравнений (11) выглядит следующим образом:

$$\begin{cases} -2\mu_{y_1 x_1} + 2\beta_{1,1}\mu_{x_1 x_1} + 2\sum\limits_{l=2}^{k}\beta_{1,l}\mu_{x_l x_1} + \lambda_1 = 0,\ (i = j = 1), \\ -2\mu_{y_i x_j} + 2\beta_{1,1}\mu_{x_i x_j} + 2\sum\limits_{l=1}^{i-1}\beta_{i,l}\mu_{x_l x_j} + 2\sum\limits_{l=i+1}^{k}\beta_{i,l}\mu_{x_l x_j} = 0,\ i,j = \overline{1,k},\ i \neq j, \\ -2\mu_{y_i x_i} + 2\beta_{1,1}\mu_{x_i x_i} + 2\sum\limits_{l=1}^{i-1}\beta_{i,l}\mu_{x_l x_i} + 2\sum\limits_{l=i+1}^{k}\beta_{i,l}\mu_{x_l x_i} + \lambda_i = \lambda_{i-1},\ i = j = \overline{2,k-1}, \\ -2\mu_{y_k x_k} + 2\beta_{1,1}\mu_{x_k x_k} + 2\sum\limits_{l=1}^{k-1}\beta_{k,l}\mu_{x_l x_k} = \lambda_{k-1},\ (i = j = k). \end{cases} \tag{12}$$

Дальнейших упрощений можно достичь путем реализации следующей схемы: 1) из уравнений, определяемых третьей и четвертой строками системы уравнений (12), находим множители Лагранжа λ_i, $i = \overline{2,k-1}$; 2) используя полученные значения множителей Лагранжа, решаем систему уравнений, определяемую первыми тремя строками системы (12), относительно элементов $\beta_{i,j}$ матрицы β. В соответствии с этой схемой четвертое уравнение из (12) представляет собой выражение для множителя Лагранжа λ_{k-1}, а третье уравнение – формулу, выражающую λ_{k-1} посредством λ_k. Последовательное применение третьего уравнения при старте с четвертого позволяет получить выражение для множителя Лагранжа λ_1,

$$\lambda_1 = -2\sum_{m=2}^{k}\mu_{y_m x_m} + 2\beta_{1,1}\sum_{m=2}^{k}\mu_{x_m x_m} + 2\sum_{m=2}^{k}\sum_{\substack{l=1,\\ l\neq m}}^{k}\beta_{m,l}\mu_{x_l x_m},\ \forall k \geq 2, \tag{13}$$

и исключить тем самым из рассмотрения третье и четвертое уравнения системы уравнений (12). В результате мы получим систему уравнений

$$\begin{cases} -2\mu_{y_1 x_1} + 2\beta_{1,1}\mu_{x_1 x_1} + 2\sum\limits_{l=2}^{k}\beta_{1,l}\mu_{x_l x_1} + \lambda_1 = 0, \\ -2\mu_{y_i x_j} + 2\beta_{1,1}\mu_{x_i x_j} + 2\sum\limits_{l=1}^{i-1}\beta_{i,l}\mu_{x_l x_j} + 2\sum\limits_{l=i+1}^{k}\beta_{i,l}\mu_{x_l x_j} = 0,\ i,j = \overline{1,k},\ i \neq j, \end{cases} \tag{14}$$

которая решается с учетом выражения (13) для λ_1. Подставив λ_1 из (13) в первое из уравнений системы (14) и выполнив упрощающие преобразования, получим систему уравнений (6), которая является окончательной для решения поставленной задачи (2), (3). Доказательство теоремы 1 окончено.

Система уравнений (6) может быть решена в два этапа: 1) приведение к стандартной форме ${}^{0,1}(AX) = B,\ A = (a_{i,j}),\ B = (b_j),\ i,j = \overline{1,k^2-k+1}$, т. е. формирование вектора неизвестных X, матрицы A СЛАУ и вектора B свободных членов; 2) решение сформированной СЛАУ стандартной программой.

Алгоритм приведения СЛАУ (6) к стандартной форме должен основываться на упорядочении неизвестных элементов $\beta_{i,j}$ матрицы β в виде вектора $X = (x_m)$. Для разработки алгоритма приведения заметим, что первое уравнение системы (6), определяемое ее первой строкой, содержит все неизвестные переменные. Поэтому формирование целесообразно начать с первого уравнения. Предлагаемый алгоритм формирования и решения СЛАУ (6) выглядит следующим образом.

1. Формируется первая строка матрицы коэффициентов A в следующем порядке: сначала формируется коэффициент $a_{1,1} = \sum_{m=1}^{k} \mu_{x_m x_m}$ при $\beta_{1,1}$, затем последующие элементы первой строки $a_{1,j}$, $j = \overline{2, k^2 - k + 1}$, путем выполнения двух вложенных циклов, определяемых суммами $\sum_{m=1}^{k} \sum_{\substack{l=1,\\ l \neq m}}^{k} \beta_{m,l} \mu_{x_l x_m}$ в (6). Эти коэффициенты определяются значениями моментов $\mu_{x_l x_m}$. Параллельно формируется $\left(2 \times (k^2 - k + 1)\right)$-матрица соответствий между элементами $\beta_{i,j}$ и элементами x_m. Номер столбца этой матрицы – значение индекса m элемента x_m, а содержимое столбца – значения двух индексов i, j элемента $\beta_{i,j}$. Наконец формируется свободный член первого уравнения $b_1 = \sum_{m=1}^{k} \mu_{y_m x_m}$.

2. Формируются последующие строки матрицы коэффициентов A. Отдельная s-я строка (отдельное уравнение) определяется фиксированными значениями индексов i, j второй строки системы уравнений (6). Здесь соблюдается тот же порядок, что и в п. 1: сначала формируется коэффициент $a_{s,1} = \mu_{x_i x_j}$ при $\beta_{1,1}$, затем – остальные элементы строки $a_{s,j}$ путем выполнения цикла в соответствии с суммой $\sum_{\substack{l=1,\\ l \neq i}}^{k} \beta_{i,l} \mu_{x_l x_j}$. Эти коэффициенты определяются значениями моментов $\mu_{x_l x_j}$. Место коэффициента в строке $a_{s,j}$ определяется с помощью сформированной в п. 1 матрицы соответствий. Остальные места строки заполняются нулями. Формируется также свободный член уравнения $b_s = \mu_{y_i x_j}$.

3. Решается сформированная в п. 1, 2 алгоритма СЛАУ, и по полученному вектору решения X формируется искомая матрица $\beta = (\beta_{i,j})$. Диагональные элементы этой матрицы приравниваются к x_1, а остальные элементы формируются с использованием полученной в п. 1 матрицы соответствий.

2. Линейное пропорциональное Прокрустово преобразование. Линейным Прокрустовым преобразованием $(k \times n)$-матрицы данных X к $(k \times n)$-матрице данных Y будем называть преобразование вида

$$Z = A + BX, \tag{15}$$

обеспечивающее определенную близость матриц Y и Z. Построенное выше линейное пропорциональное преобразование (5) вектора x к вектору y позволяет формально построить линейное пропорциональное Прокрустово преобразование с критерием близости, определяемым постановкой задачи (2)–(4). Для этого первоначально отметим, что преобразование (5) справедливо также для наблюдений x_j и y_j векторов x и y, а также для матриц $X = (x_j) = (x_{i,j})$, $Y = (y_j) = (y_{i,j})$, столбцами которых являются векторы наблюдений x_j, y_j, $i = \overline{1,k}$, $j = \overline{1,n}$, где n – число наблюдений (объем выборки). В этом случае вместо векторного преобразования (5) мы будем иметь следующее матричное преобразование:

$$Z = \bar{\nu}_y + {}^{0,1}\left(\beta(X - \bar{\nu}_x)\right),$$

где $\bar{\nu}_x = (\nu_x, \nu_x, ..., \nu_x)$, $\bar{\nu}_y = (\nu_y, \nu_y, ..., \nu_y)$ – матрицы, содержащие n одинаковых столбцов ν_x и ν_y соответственно, и β – матрица, определяемая путем решения СЛАУ (6). Далее, неизвестные теоретические моменты ν_x, ν_y, $\mu_{x^2} = (\mu_{x_i x_j})$, $\mu_{yx} = (\mu_{y_i x_j})$ заменим их выборочными оценками $\hat{\nu}_x$, $\hat{\nu}_y$, $\hat{\mu}_{x^2} = (\hat{\mu}_{x_i x_j})$, $\hat{\mu}_{yx} = (\hat{\mu}_{y_i x_j})$ соответственно, полученными по матрицам наблюдений $X = (x_j) = (x_{i,j})$, $Y = (y_j) = (y_{i,j})$:

$$\hat{\nu}_x = \frac{1}{n}\sum_{j=1}^{n} x_j \, , \hat{\nu}_y = \frac{1}{n}\sum_{j=1}^{n} y_j \, ,$$

$$\hat{\mu}_{x^2} = (\hat{\mu}_{x_i x_j}) = \frac{1}{n}\sum_{j=1}^{n} {}^{0,0}(x_j)^2, \hat{\mu}_{yx} = (\hat{\mu}_{y_i x_j}) = \frac{1}{n}\sum_{j=1}^{n} {}^{0,0}(y_j x_j).$$

В итоге мы получим следующее преобразование матрицы данных $X = (x_j) = (x_{i,j})$ в матрицу данных $Z = (z_j) = (z_{i,j})$:

$$Z = \hat{\hat{\nu}}_y + {}^{0,1}(\hat{\beta}(X - \hat{\hat{\nu}}_x)), \tag{16}$$

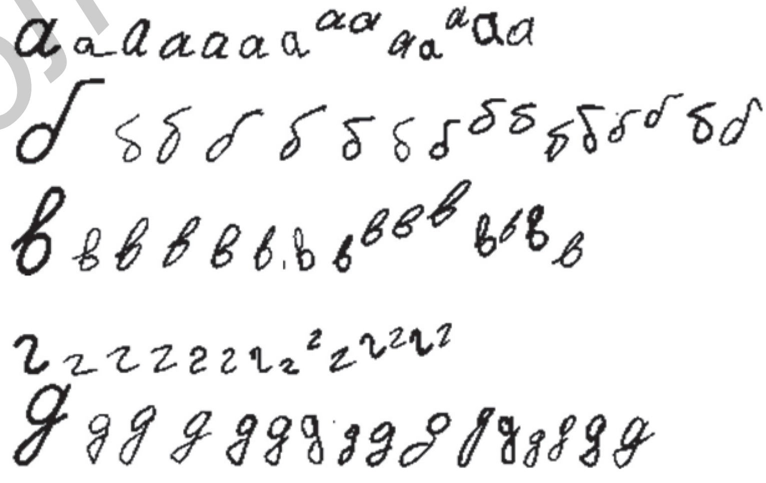
где $\hat{\hat{\nu}}_x = (\hat{\nu}_x, \hat{\nu}_x, ..., \hat{\nu}_x)$, $\hat{\hat{\nu}}_y = (\hat{\nu}_y, \hat{\nu}_y, ..., \hat{\nu}_y)$ – матрицы, содержащие по n одинаковых столбцов $\hat{\nu}_x$ и $\hat{\nu}_y$ соответственно; $\hat{\beta}$ – матрица, определяемая путем решения СЛАУ (6) при замене $\mu_{x^2} = (\mu_{x_i x_j})$ на $\hat{\mu}_{x^2} = (\hat{\mu}_{x_i x_j})$ и $\mu_{yx} = (\mu_{y_i x_j})$ на $\hat{\mu}_{yx} = (\hat{\mu}_{y_i x_j})$. Преобразование (16) будем называть линейным пропорциональным Прокрустовым преобразованием.

3. Распознавание рукописных букв. Предложенное линейное пропорциональное Прокрустово преобразование (16) было запрограммировано в виде m-файла-функции Матлаб и использовано при распознавании рукописных букв наряду с другими преобразованиями. Набор распознаваемых букв содержал 478 кириллических рукописных строчных букв (примерно по 15 реализаций каждой), написанных различными людьми. Ниже представлен фрагмент указанного набора букв (рисунок). Он собран из графических файлов букв. Разброс последних по высоте обусловлен тем, что файлы имеют различные размеры. В качестве эталона использовались буквы прописей, представленные первыми в каждой строке. Распознаваемая буква подгонялась к каждой букве эталона с помощью Прокрустова преобразования. Результатом распознавания считался эталон, на котором достигнуто минимальное Прокрустово расстояние. Процесс распознавания был организован так, как описано в работах [3, 7]. Результаты распознавания приведены в таблице.

Результаты распознавания 478 русских рукописных строчных букв кириллицы с помощью различных преобразований при эталонах-прописях

Применяемое при распознавании преобразование	Линейное Прокрустово преобразование [2, 3]	Простое симметричное Прокрустово преобразование [8]	Ортогональное Прокрустово преобразование [1]	Пропорциональное Прокрустово преобразование
Число и процент правильных распознаваний	260 (54,4 %)	285 (59,6 %)	284 (59,4 %)	294 (61,5 %)

Из таблицы видно, что использование предложенного пропорционального Прокрустова преобразования позволяет повысить точность распознавания на 7 % по сравнению с линейным Прокрустовым преобразованием и на 2 % по сравнению с ортогональным Прокрустовым преобразованием.



Фрагмент набора букв для распознавания

Более точным по сравнению с ортогональным в данном случае оказалось также простое симметричное Прокрустово преобразование, предложенное в работе [8].

Таким образом, предложенное пропорциональное Прокрустово преобразование представляет собой реальную альтернативу известному ортогональному Прокрустову преобразованию в задаче распознавания рукописных букв.

П р и л о ж е н и е. Следом $2q$-мерной матрицы $V=(v_{i,j})$, $i=(i_1,i_2,...,i_q)$, $j=(j_1,j_2,...,j_q)$, индексы мультииндексов i и j которой пробегают множества одних и тех же значений, называется скалярная величина вида

$$\operatorname{tr} V=\sum_i v_{i,i}\,.$$

Т е о р е м а 2 *(о производной следа многомерной матрицы). Если* $V=(v_{i,j})$, $i=(i_1,i_2,...,i_q)$, $j=(j_1,j_2,...,j_q)$, *– $2q$-мерная матрица, имеющая след, то*

$$\frac{d}{dV}\operatorname{tr} V=E(0,q). \tag{17}$$

Д о к а з а т е л ь с т в о. По определению многомерно-матричной производной [6] получаем

$$\frac{d}{dV}\operatorname{tr} V=\left(\frac{\partial}{\partial v_{k,l}}\sum_i v_{i,i}\right)=\left(\sum_i \frac{\partial v_{i,i}}{\partial v_{k,l}}\right)=$$

$$=\left(\sum_i \begin{cases}1, & i=k=l\\ 0, & \text{иначе}\end{cases}\right)=\left(\begin{cases}1, & k=l\\ 0, & \text{иначе}\end{cases}\right)=Z=(z_{k,l})=E(0,q).$$

Доказательство теоремы 2 окончено.

Список использованной литературы

1. *Crosilla, F.* Procrustes Analysis and Geodetic Sciences / Fabio Crosilla // Technical report. – Stuttgart: Univ. of Stuttgart, Dept. of Geodesy and Geoinformatics. – 1999. – Pt. 1. – P. 69–78.

2. *Муха, В. С.* Линейное Прокрустово преобразование двухмерных матриц / В. С. Муха // Информатика. – 2010. – № 3. – С. 97–102.

3. *Муха, В. С.* Линейное Прокрустово преобразование в распознавании букв / В. С. Муха // Автоматика и вычисл. техника. – 2012. – № 3. – С. 36–48.

4. *Муха, В. С.* Распознавание букв на основе Прокрустова преобразования / В. С. Муха // Информационные технологии и системы – 2012 (ИТС 2012): материалы Междунар. науч. конф. (БГУИР, Минск, Беларусь, 24 окт. 2012) / редкол.: Л. Ю. Шилин [и др.]. – Минск, 2012. – С. 266–267.

5. *Муха, В. С.* Пропорциональное Прокрустово преобразование в распознавании рукописных цифр и знаков / В. С. Муха, А. Н. Кузьков // Информационные технологии и системы 2014 (ИТС 2014): материалы Междунар. науч. конф. (БГУИР, Минск, Беларусь, 29 окт. 2014) / редкол.: Л. Ю. Шилин [и др.]. – Минск, 2014. – С. 290–291.

6. *Муха, В. С.* Анализ многомерных данных / В. С. Муха. – Минск: Технопринт, 2004.

7. *Муха, В. С.* Прокрустово преобразование в распознавании рукописных букв / В. С. Муха, А. Н. Кузьков // Информационные технологии и системы – 2013 (ИТС 2013): материалы Междунар. науч. конф. (БГУИР, Минск, Беларусь, 23 окт. 2013) / редкол.: Л. Ю. Шилин [и др.]. – Минск, 2013. – С. 298–299.

8. *Муха, В. С.* Линейное простое симметричное Прокрустово преобразование / В. С. Муха, А. Н. Кузьков // Докл. БГУИР. – 2015. – № 5 (91). – С. 25–28.

Поступила в редакцию 03.09.2015