MATRIX FEATURE VECTORS AND HU MOMENTS IN GESTURE RECOGNITION

Volodymyr Donchenko, Andrew Golik

***Abstract**: This paper covers usage of matrix feature vectors and Hu moments in recognition of tactile sign language. The paper also provides comparative characteristic of both approaches and a variant of formation of feature vectors in matrix form. It is suggested to use orthogonal and ellipsoidal compliance distances for matrix feature vectors and numerical intervals for Hu moments.*

***Keywords**: gesture recognition, Hu moments, orthogonal projectors, ellipsoidal distance, SVD – decomposition, pseudoinverse.*

***ACM Classification Keywords**: I.2 Artificial Intelligence, I.4 Image Processing and Computer Vision, I.5 Pattern Recognition, G.1.3 Numerical Linear Algebra.*

Introduction

This article draws parallels between usage of Hu moments and matrix feature vectors in gesture recognition. Specific case of the mentioned task was chosen for implementation and testing: finger recognition of sign language. Hu moments are well-known numeric characteristics that can be obtained for image of gesture and effectively used for gesture recognition. They are so wide-used because Hu moments are invariant under translation, changes in scale and rotation. However, such power requires corresponding level of responsibility. We usually consider all the moments at the same time as feature vector that can be used for clustering. This approach has a lot of leaks which are covered in the paper.

In order to find more stable and effective solution matrix feature vectors are suggested. Usage of matrices as representatives of the object which is analyzed is “natural” technique. Gestures are presented with images (or sequence of images) that in early stages of processing of input data are captured from a webcam or other recording device. A variant of conversion of the images to matrices is suggested in the article.

Two variants of compliance distances are suggested, namely ellipsoidal and orthogonal distances. Ellipsoidal distance is based on a "minimal ellipse" that "covers" learning sample of class. Orthogonal distance is based on Cartesian grouping operators and orthogonal projectors.

Clustering with usage of compliance distances that are based on pseudoinverse and SVD-decomposition can be successfully applied to numeric vectors. However, as mentioned above learning sample consists of matrices. One of the main purposes of the research was to transfer properties of pseudoinverse and SVD-decomposition to the space of matrix feature vectors.

Results of recognition program that was implemented using C# and EmguCV environments justify an introduction of mentioned approaches, especially, compliance distances that are based on orthogonal projectors.

Overview of Hu moments

Image moment is a certain particular weighted average of the image pixels' intensities, or a function of such moments, usually chosen to have some attractive property or interpretation. Simple properties of the image which are found via image moments include area, its centroid, and information about its orientation.

For a 2D continuous function $f(x,y)$ the moment of order $(p+q)$ is defined as

$$M_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^p y^q f(x,y) dxdy$$

for $p,q = 0,1,2,\ldots$ Adapting this to greyscale image with pixel intensities $I(x,y)$, raw image moments M_{ij} are calculated by

$$M_{ij} = \sum \sum x^i y^j I(x,y)$$

In some cases, this may be calculated by considering the image as a probability density function, i.e., by dividing the above by

$$\sum_x \sum_y I(x,y)$$

A uniqueness theorem (Hu [1962]) states that if $f(x,y)$ is piecewise continuous and has nonzero values only in a finite part of the x,y plane, moments of all orders exist, and the moment sequence (M_{pq}) is uniquely determined by $f(x,y)$. Conversely, (M_{pq}) uniquely determines $f(x,y)$. In practice, the image is summarized with functions of a few lower order moments.

Simple image properties derived via moments include:

- Area (for binary images) or sum of grey level: M_{00}
- Centroid: $\{x,y\} = \{M_{10}/M_{00}, M_{01}/M_{00}\}$

Central moments are defined as

$$M_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (x-\bar{x})^p (y-\bar{y})^q f(x,y) dxdy$$

where $\bar{x} = \frac{M_{10}}{M_{00}}$ and $\bar{y} = \frac{M_{01}}{M_{00}}$ are the components of the centroid.

If $f(x,y)$ is a digital image, then the previous equation becomes

$$M_{pq} = \sum_x \sum_y (x-\bar{x})^p (y-\bar{y})^q f(x,y)$$

$$\mu_{pq} = \sum_m^p \sum_n^q \binom{p}{m}\binom{q}{n} (-\bar{x})^{(p-m)} (-\bar{y})^{(q-n)} M_{mn}$$

Central moments are translational invariant.

Information about image orientation can be derived by first using the second order central moments to construct a covariance matrix.

$$\mu'_{20} = \mu_{20}/\mu_{00} = M_{20}/M_{00} - \bar{x}^2,\ \mu'_{02} = \mu_{20}/\mu_{00} = M_{02}/M_{00} - \bar{y}^2,\ \mu'_{11} = \mu_{11}/\mu_{00} = M_{11}/M_{00} - \bar{x}\bar{y}$$

The covariance matrix of the image $I(x,y)$ is now

$$\operatorname{cov}[I(x,y)] = \begin{bmatrix} \mu'_{20}\mu'_{11} \\ \mu'_{11}\mu'_{02} \end{bmatrix}.$$

The eigenvectors of this matrix correspond to the major and minor axes of the image intensity, so the orientation can thus be extracted from the angle of the eigenvector associated with the largest eigenvalue. It can be shown that this angle Θ is given by the following formula:

$$\Theta = \frac{1}{2}\arctan\left(\frac{2\mu'_{11}}{\mu'_{20} - \mu'_{02}}\right)$$

The above formula holds as long as:

$$\mu'_{11} \neq 0$$

The eigenvalues of the covariance matrix can easily be shown to be

$$\lambda_i = \frac{\mu'_{20} + \mu'_{02}}{2} \pm \frac{\sqrt{4\mu'^2_{11} + (\mu'_{20} - \mu'_{02})}^2}{2}$$

and are proportional to the squared length of the eigenvector axes. The relative difference in magnitude of the eigenvalues is thus an indication of the eccentricity of the image, or how elongated it is. The eccentricity is

$$\sqrt{1 - \frac{\lambda_2}{\lambda_1}}$$

Moments n_{ij} where $i + j \geq 2$ can be constructed to be invariant to both translation and changes in scale by dividing the corresponding central moment by the properly scaled (00)th moment, using the following formula.

$$n_{ij} = \frac{\mu_{ij}}{\mu_{00}^{1+\frac{i+j}{2}}}$$

It is possible to calculate moments which are invariant under translation, changes in scale, and also rotation. Most frequently used are the Hu set of invariant moments:[6]

$$I_1 = \eta_{20} + \eta_{02}$$

$$I_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$

$$I_3 = (\eta_{30} + 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$

$$I_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

$$I_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} - \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]$$

$$I_6 = (\eta_{20} - \eta_{02})\left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$

$$I_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]$$

The first one, I1, is analogous to the moment of inertia around the image's centroid, where the pixels' intensities are analogous to physical density. The last one, I7, is skew invariant, which enables it to distinguish mirror images of otherwise identical images.

A general theory on deriving complete and independent sets of rotation invariant moments was proposed by J. Flusser[7] and T. Suk.[8] They showed that the traditional Hu's invariant set is not independent nor complete. I3 is not very useful as it is dependent on the others. In the original Hu's set there is a missing third order independent moment invariant:

$$I_8 = \eta_{11}\left[(\eta_{30} + \eta_{12})^2 - (\eta_{03} + \eta_{21})^2\right] - (\eta_{20} - \eta_{02})(\eta_{30} + \eta_{12})(\eta_{03} + \eta_{21})$$

Matrix feature vectors

First stage of gesture recognition problem consists of capturing images from a webcam or other recording device, followed by finding and highlighting on the resulting image hand and its contour. This contour gives fairly complete information that can be used for gesture identification.

There are several ways to analyze a contour of hand, for example, as series of interrelated points. In addition, there are a number of numerical characteristics that can be calculated for the contour: moments, Freeman chains etc. We are going to talk about representation of gesture contour in matrix form. Transition to matrix form begins with finding the smallest rectangle covering a contour of hand on an image.

Having its coordinates, we can cut it from an image and convert into binary matrix. However, standardization problem of dimension of such matrices is urgent because it depends on many factors: size of hand of a person, a distance from hand to recording device, etc. Possible solution of this problem is a construction of “characteristic” matrix: capturing images of contour of hand and its subsequent compression or stretching to standard size with conversion into matrix form according to certain rules. However, if we consider gesture recognition problem specific variant of scaling is required. An example that clearly demonstrates a need for changes in the above mentioned algorithm of standardization is shown in Figure 3.

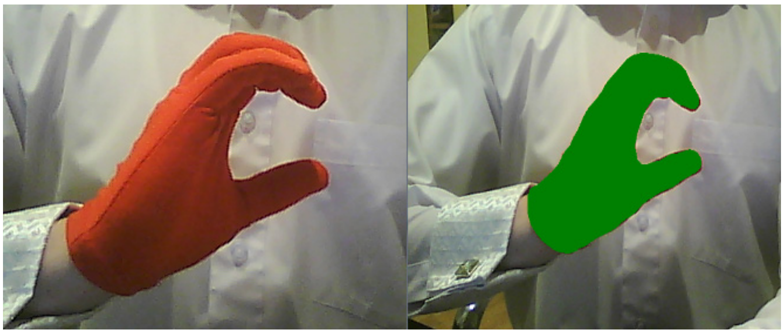


Figure 1. Image of gesture that is captured from a webcam. A contour of hand is found and highlighted

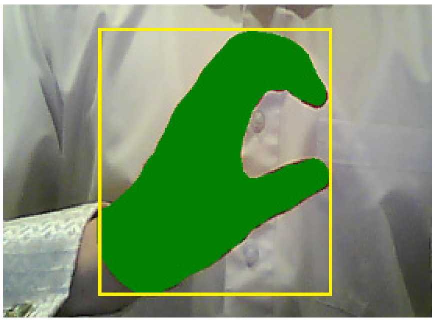


Figure 2. The smallest rectangle covering a contour of hand

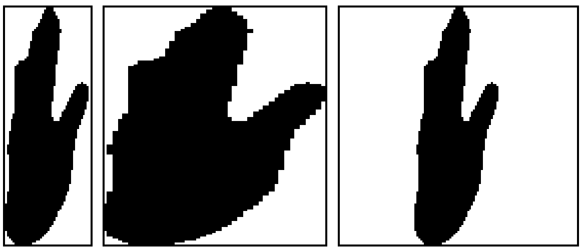


Figure 3. Different variants of standardization of an image.

There are 3 parts of Figure 3: the first - minimal rectangle covering a gesture, the other two - variants of its standardization. Suppose that square of certain size was chosen as a standard. In this case, after stretching an image we will get results that are presented in the second part of Figure 3. It is not difficult to see that in this case an image of a gesture largely lost its informative value, because the ratio of width and height, which is important in this problem, was changed. More correct approach is illustrated in the third part of Figure 3. In this case additional empty areas were placed on the left and right from the image. The size of these areas is identical and found in such way that a resulting image conforms to the standards.

It is suggested to do a transition from an image to the matrix on the next stage. We remind that RGB is a format of presentation of color, as a combination of red, green and blue colors. Having results of experiments we set the legitimate values of RGB, which allow to make decision: whether a pixel should be examined as meaningful or not. The transformation of an image consists of replacement of pixels which satisfy the set of legitimate values of RGB by 1 and all other by 0. Finally we get a matrix that consists of 0 and 1. The matrix can be called a "characteristic" matrix. Its image can be reproduced in a black-an-white form which is natural for binary matrices.

The standardized «characteristic» matrices for the images of gestures can be used for recognition of signs of tactile language. A binary characteristic matrix is obtained as the result converting process.

Ellipsoidal and orthogonal compliance distances

After forming feature vectors on the stage of clustering there is a necessity for comparison of the vectors, establishment of the so-called compliance distance between them. Possibility of usage of ellipsoidal and orthogonal distances is considered in the article.

The main feature of the mentioned distances is that while training the system, they work not with one etalon, but with a set of etalons (for the different environmental conditions).

Ellipsoidal distance is built by facilities of pseudoinverse for different variants of linear operators. Such distance leans against conception of «minimum ellipses of grouping». Actually, we talk about ellipses that «cover» each of training sets by a «minimum» and «optimum» rank. Ellipsoidal distance is built for matrices as matrices of linear operators between matrix Euclidian spaces by facilities of pseudoinverse for the mentioned spaces. They are implemented, as well as in the case of vector Euclidian spaces, through the so-called «groupings operators» of theory of pseudoinverse. Such operators are determined after the matrix of operator A that is operator between vector Euclidian spaces, and is defined by expressions:

$$R(A) = A^{+}A^{+T}, R(A^{T}) = (A^{T})^{+}(A)^{T+T} = A^{+T}A^{+}$$

The principle role of grouping operators is that they allow us to build the «minimum ellipses of grouping»: ellipsoids which contain all vectors of set $a_k, k = \overline{1,n}$ and are optimum in certain sense. Optimum lies in following: all axis of the ellipse are formed by the orthonormal set of vectors, sum of squares of projections on which is maximal, and the squares of lengths of proper axis coincide with the proper sums of squares of projections. More precisely next four theorems have place [4].

Theorem 1 For an arbitrary set of vectors $a_k \in R^m, k = \overline{1,n}$, solution of optimization problem of search of maximum sum of squares of projections on subspace that is formed by the normalized vector $u \in R^m : \|u\| = 1$ is a vector u_1 from singularity (u_1, λ_1^2) of singular decomposition of matrix $A = (a_1 \vdots \ldots \vdots a_n)$:

$$u_1 = \arg \min_{u \in R^m : \|u\|=1} \sum_{k=1}^{r} \| \Pr_u a_k \|^2$$

$$\min_{u \in R^m : \|u\|=1} \sum_{k=1}^{r} \| \Pr_u a_k \|^2 = \lambda_1^2$$

Theorem 2 For arbitrary set of vectors $a_k \in R^m, k = \overline{1,n}$, solution of optimization problem of search of maximum sum of squares of projections on subspace that is formed by normalized vector $u \in R^m : \|u\| = 1$ is a vector u_1 from singularity (u_1, λ_1^2) of singular decomposition of matrix $A = (a_1 \vdots \ldots \vdots a_n)$:

$$u_k = \arg \min_{u \in R^m : \|u\|=1, u \perp L(u_1, \ldots, u_k)} \sum_{k=1}^{r} \| \Pr_u a_k \|^2$$

$$\min_{u \in R^m : \|u\|=1, u \perp L(u_1, \ldots, u_k)} \sum_{k=1}^{r} \| \Pr_u a_k \|^2 = \lambda_{k+1}^2$$

$$k = \overline{1, r-1},$$

where $(u_k, \lambda_k^2), k = \overline{1,r}$ as well as in the previous theorem of singularity of singular decomposition of matrix which is formed from the elements of the researched set of vectors.

Theorem 3 For arbitrary set of vectors $a_k \in R^m, k = \overline{1,n}$

$$a_k^T R(A^T) a_k \le r_{\max}^2 < r$$

$$r_{\max}^2 = \max_{k=\overline{1,n}} a_k^T \, R(A^T) a_k \,,$$

Where, as well as in two previous theorems, A is a matrix that is formed from the vectors of a set as its columns. Ellipsoid of theorem 3 groups the vectors of set according to the central location of the ellipse of grouping: based on an ellipse which has center at origin. In practical applications center of ellipse is mean value $\bar{a}$ of elements from the set:

$$\bar{a} = \frac{1}{n} \sum a_k$$

In this case a grouping operator is built based on a matrix $\tilde{A}$ which is formed from centered average vectors from the set $\tilde{a}_k : \tilde{a}_k = a_k - \bar{a}, k = \overline{1,n}$. Consequently following theorem has place.

Theorem 4 For arbitrary set of vectors $a_k \in R^m, k = \overline{1,n}$ we have following inequalities

$$(a_k - \bar{a})^T R(\tilde{A}^T)(a_k - \bar{a}) \le \tilde{r}_{\max}^2 \le r, k = \overline{1,n}$$

$$r_{\max}^2 = \max_{k=\overline{1,n}} \tilde{a}_k^T R(\tilde{A}^T) \tilde{a}_k$$

As a set of vectors the training sets of classes are used $Kl_l, l = \overline{1,L}$. As compliance distances (namely their squares): functional $\rho^2(x, Kl_l), x \in R^m, l = \overline{1,L}$ according to minimum value of which sorting is performed, - it is possible to use the minimum ellipses of grouping. It means that compliance distances are determined as following:

$$\rho^2(x, Kl_l) = (x - \bar{a}_l)^T \frac{R(\tilde{A}_1^T)}{\tilde{r}_{1\max}^2}(x - \bar{a}_l), x \in R^m \,, \; l = \overline{1,L}$$

Such ellipsoidal distance is used for characteristic matrices.

Together with ellipsoidal compliance distance orthogonal distance is offered in the article. It gives ability to carry properties of pseudoinverse and SVD– decomposition in case of matrix feature vectors.

$R^{(m\times n),K}$ is Euclidian space $m \times n$ of matrix corteges of length K $\alpha = (A_1 \vdots \ldots \vdots A_K) \in R^{(m\times n),K}$ with «natural» component-wise scalar multiplication:

$$(\alpha, \beta) = \sum_{k=1}^{K} (A_k, B_k)_{tr} = \sum_{k=1}^{K} tr A_k^T B_k$$

$$\alpha = (A_1 \vdots \ldots \vdots A_K), \beta = (B_1 \vdots \ldots \vdots B_K) \in R^{(m\times n),K}$$

$\wp_\alpha : R^K \to R^{m\times n}$ linear operator between corresponding Euclidian spaces, that is set by a matrix cortege $\alpha = (A_1 \vdots \ldots \vdots A_K) \in R^{(m\times n),K}$ and determined by matrix cortege operations according to expression:

$$\wp_\alpha y = \sum_{k=1}^{K} y_k A_k, \alpha = (A_1 \vdots \ldots \vdots A_K) \in R^{(m\times n),K}, y = \begin{pmatrix} y_1 \\ \cdots \\ y_K \end{pmatrix} \in R^K$$

Theorem 5 [5] Conjugate $\wp_\alpha^*$ of the operator $\wp_\alpha : R^K \to R^{m\times n}$ is a linear operator, which obviously, operates in reverse to $\wp_\alpha$ direction: $\wp_\alpha^* : R^{m\times n} \to R^K$ and is determined by expression:

$$\wp_\alpha^* X = \begin{pmatrix} tr A_1^T X \\ \cdots \\ tr A_K^T X \end{pmatrix}$$

Proof

Indeed,

$$(\wp_\alpha y, X)_{tr} = \left(\sum_{k=1}^{K} y_k A_k, X\right)_{tr} = \sum_{k=1}^{K} y_k (A_k, X)_{tr} = \sum_{k=1}^{K} y_k \left(trA_k^T X\right) = \left(y, \begin{pmatrix} trA_1^T X \\ \cdots \\ trA_K^T X \end{pmatrix}\right)$$

This proves the theorem.

Theorem 6 [5] Multiplication of two operators is a linear operator $\wp_\alpha^* \wp_\alpha : R^K \to R^K$ which is given by a matrix (we will identify it with the operator), which is determined by expression:

$$\wp_\alpha^* \wp = \begin{pmatrix} trA_1^T A_1, ..., trA_1^T A_n \\ \cdots \\ trA_n^T A_1, ..., trA_n^T A_n \end{pmatrix} \quad (1)$$

Notice that matrix that is defined by expression (1) is the matrix of Gramm of elements $A_1, ..., A_K$ of matrix cortege $\alpha = (A_1 \vdots ... \vdots A_K)$, that specifies operator $\wp_\alpha$.

Proof

Indeed,

$$\wp_\alpha^* \wp_\alpha y = \wp_\alpha^*(\wp_\alpha y) = \begin{pmatrix} trA_1^T \sum_{i=1}^{n} A_i y_i \\ \cdots \\ trA_n^T \sum_{i=1}^{n} A_i y_i \end{pmatrix} = \begin{pmatrix} tr\sum_{i=1}^{n} A_1^T A_i y_i \\ \cdots \\ tr\sum_{i=1}^{n} A_n^T A_i y_i \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{n} trA_1^T A_i y_i \\ \cdots \\ \sum_{i=1}^{n} trA_n^T A_i y_i \end{pmatrix} = \begin{pmatrix} trA_1^T A_1, ..., trA_1^T A_n \\ \cdots \\ trA_n^T A_1, ..., trA_n^T A_n \end{pmatrix} \begin{pmatrix} y_1 \\ \cdots \\ y_n \end{pmatrix} = (tr(A_i^T A_j))y$$

This proves the theorem.

A singular decomposition for a matrix (1) is obvious: it is symmetric and non-negatively defined matrix. It is determined by the set of singularities $(v_i, \lambda_i^2), i, j = \overline{1,r}$: by the orthonormal set of vectors $\| v_i \| = 1, v_i \perp v_j, i \neq j; i, j = \overline{1,r}; \lambda_1 > \lambda_2 > ... > \lambda_r > 0$ which are own for an operator $\wp_\alpha^* \wp_\alpha : R^K \to R^K$: $\wp_\alpha^* \wp_\alpha v_i = \lambda_i^2 v_i, i = \overline{1,r}$. Defined by singularities $(v_i, \lambda_i^2), i = \overline{1,r}$ matrices $U_i \in R^{m \times n} : U_i = \frac{1}{\lambda_i} \wp_\alpha v_i, i = \overline{1,r}$ are the elements of set of singularities $(U_i, \lambda_i^2), i = \overline{1,r}$ of the operator $\wp_\alpha \wp_\alpha^*$. Singular decomposition of cortege operator: singularities of two operators: $\wp_\alpha^* \wp_\alpha, \wp_\alpha \wp_\alpha^*$, determine the singular decomposition of operator $\wp_\alpha$.

Theorem 7 [5] (singular decomposition of cortege operator)

$$\wp_\alpha = \sum_{k=1}^{K} \lambda_k U_k v_k^T.$$

Variant of singular decomposition: taking into consideration the expression $U_i \in R^{m \times n} : U_i = \frac{1}{\lambda_i} \wp_\alpha v_i, i = \overline{1,r}$ n and its investigation, we have

$$\wp_\alpha = \sum_{k=1}^{K} \lambda_k U_k v_k^T = \sum_{k=1}^{K} (\wp_\alpha v_k) v_k^T$$

Remark of general character: the general variant of the theorem about singular decomposition is needed. This statement should touch general Euclidian spaces. It needs to be formulated for linear operators on general Euclidian spaces.

Theorem 8 [5] For an arbitrary linear operator $\wp_E : E_1 \to E_2$ on the pair of Euclidian spaces $(E_i, (\,,\,)_i), i = 1,2$ there is a set of singularities $(v_i, \lambda_l^2), (u_i, \lambda_l^2) i = \overline{1,r}, r = rank \wp_E$ of operators $\wp_E^* \wp$, $\wp \wp_E^*$ accordingly with the general set of own numbers $\lambda_l^2, i = \overline{1,r}$ that

$$\wp_E x = \sum_{i=1}^{r} u_i \lambda (v_i, x)_1, \wp_E^* y = \sum_{i=1}^{r} v_i \lambda (u_i, y)_2$$

In addition following expressions have place:

$$u_i = \lambda_l^{-1} \wp v_i, i = \overline{1,r}$$

$$v_i = \lambda_l^{-1} \wp_E^* u_i, i = \overline{1,r}$$

Basic operators of PDO theory are for cortege operators: a pseudoinverse by svd-decomposition.

According to svd-determination, PDO of cortege operator is set by following expression [5]:

$$\wp_\alpha^+ = \sum_{k=1}^{K} \lambda^{-1} v_k \left(U_k, \cdot\right)_{tr} = \sum_{k=1}^{K} \lambda^{-2} v_k \left(\wp_\alpha v_k, \cdot\right)_{tr}$$

The orthogonal projectors of base subspaces of operator and, accordingly, - grouping operators are determined after svd-presentation of cortege operator in standard way.

Theorem 9 Operators marked as $P(\wp_\alpha^*), P(\wp_\alpha)$ and determined by expressions:

$$P(\wp_\alpha^*) = \sum_{k=1}^{r} U_k \left(U_k, \cdot\right)_{tr}$$

$$P(\wp_\alpha) = \sum_{k=1}^{r} v_k \left(v_k, \cdot\right) = \sum_{k=1}^{r} v_k v_k^T$$

are orthogonal projectors $P_{L_{\wp_\alpha}}, P_{L_{\wp_\alpha^*}}$ on subspaces $L_{\wp_\alpha}$, $L_{\wp_\alpha^*}$ of possible values of operators $\wp_\alpha$, $\wp_\alpha^*$ accordingly:

$$P(\wp_\alpha^*) = P_{L_{\wp_\alpha}}, P(\wp_\alpha) = P_{L_{\wp_\alpha^*}}$$

These subspaces are the linear shells of the corresponding orthonormal sets:

$$L_{\wp_\alpha} = L(U_1, ..., U_r), \;\; L_{\wp_\alpha^*} = L(v_1, ..., v_r)$$

Proof

Proof is the same as in the case of linear operators between Euclidian spaces of numerical vectors: symmetry and idempotence is simply checked up for both operators. Similarly obvious are assertions that $U_k \in L_{\wp_\alpha}$, $v_k \in L_{\wp_\alpha^*}$, and consequently from reasoning of dimension $L_{\wp_\alpha} = L(U_1, ..., U_r)$, $L_{\wp_\alpha^*} = L(v_1, ..., v_r)$. In addition, as follows from determination $P_{L_{\wp_\alpha}}, P_{L_{\wp_\alpha^*}}$ the last spaces are spaces of possible values for them accordingly. Finally, note, that subspace on which an orthogonal projector carries out the orthogonal projection can be described, in particular, as a space of possible values for it.

Theorem 10 Operators $Z(\wp_\alpha^*), Z(\wp_\alpha)$ which are complements to the identical operator of orthogonal projectors $P(\wp_\alpha^*), P(\wp_\alpha)$ accordingly:

$$Z(\wp_\alpha^*) X = X - P(\wp_\alpha^*) X, \quad Z(\wp_\alpha) = E_K - P(\wp_\alpha),$$

are orthogonal projectors on the kernels of operators accordingly.

Proof

Firstly, proof follows from the fact that for $\wp_\alpha^*$, $\wp_\alpha$ each of operators $Z(\wp_\alpha^*), Z(\wp_\alpha)$ is symmetric and idempotent. In addition they are orthogonal projectors on the orthogonal adding to subspaces

$L_{\wp_\alpha} = L(U_1,...,U_r)$, $L_{\wp_\alpha^*} = L(v_1,...,v_r)$ accordingly. Namely, these orthogonal complements are the kernels of operators $\wp_\alpha^*, \wp_\alpha$ accordingly.

Theorem 11 Square of distance $\rho^2(X, L_{\wp_\alpha})$ from arbitrary $m \times n$ matrix X to linear subspace $L_{\wp_\alpha}$ that is the set of possible values of cortege operator $\wp_\alpha$ is given by formula:

$$\rho^2(X, L_{\wp_\alpha}) = (X, Z(\wp_\alpha^*)X)_{tr} = \| X \|^2 - \sum_{k=1}^{r} (X, U_k)_{tr}^2$$

Proof

Indeed,

$\rho^2(X, L_{\wp_\alpha}) = \| X_{L_{\wp_\alpha}^\perp} \|^2$ in decomposition $X = X_{L_{\wp_\alpha}} + X_{L_{\wp_\alpha}^\perp}$ by decomposition $R^{m\times n} = L_{\wp_\alpha} + L_{\wp_\alpha^*}$. Obviously, $X_{L_{\wp_\alpha}^\perp} = Z(\wp_\alpha^*)X$ so:

$$\rho^2(X, L_{\wp_\alpha}) = \| X_{L_{\wp_\alpha}^\perp} \|^2 = \| Z(\wp_\alpha^*)X \|_{tr}^2 = \left(Z(\wp_\alpha^*)X, Z(\wp_\alpha^*)X_{tr} \right) = \left(X, Z(\wp_\alpha^*)Z(\wp_\alpha^*)X_{tr} \right)_{tr} = \left(X, Z(\wp_\alpha^*)X_{tr} \right)_{tr}$$

As an orthonormal set $U_i, i = \overline{1,r}$ is an orthonormal base in $L_{\wp_\alpha} = L(U_1,...,U_r)$ and $(X, U_i)_{tr}, i = \overline{1,r}$ is the co-ordinates of decomposition $X_{L_{\wp_\alpha}}$ by this orthonormal base, then $\| X_{L_{\wp_\alpha}} \|^2 = \sum_{i=1}^{r} (X, U_i)_{tr}^2$.

It remains to notice that according to the theorem of Pythagoras in an abstract variant $\| X \|^2 = \| X_{L_{\wp_\alpha}} \|^2 + \| X_{L_{\wp_\alpha}^\perp} \|^2$, and consequently:

$$\| X_{L_{\wp_\alpha}^\perp} \|^2 = \| X \|^2 - \| X_{L_{\wp_\alpha}} \|^2 = \| X \|^2 - \sum_{k=1}^{r} (X, U_k)_{tr}^2$$

The theorem is well-proven.

Theorem 12 A square of distance $\rho^2(X, L)$ of arbitrary $m \times n$ matrix X to linear subspace $L = L(A_1,...,A_K)$, which is the linear hull of set $m \times n$ matrices $A_1,...,A_K$ is determined by formula:

$$\rho^2(X, L) = \rho^2(X, L_{\wp_\alpha}) = (X, Z(\wp_\alpha^*)X)_{tr} = \| X \|^2 - \sum_{k=1}^{r} (X, U_k)_{tr}^2$$

for a cortege operator $\wp_\alpha$, formed by a set $A_1,...,A_K$: $\wp_\alpha = (A_1,...,A_K)$.

Proof

Proof follows from the fact that subspaces $L = L(A_1,...,A_K)$ and $L_{\wp_\alpha}$ coincide between itself.

Theorem 13 A square of distance $\rho^2(X, \Gamma(a, L))$ of arbitrary $m \times n$ matrix X to the hyper plane $\Gamma(\bar{a}, L)$:

$$\bar{a} = \frac{1}{K}\sum_{k=1}^{K} A_k, L = L(\tilde{A}_1,...,\tilde{A}_K), \tilde{A}_k = A_k - \bar{a}, k = \overline{1,K},$$

formed by set of $m \times n$ matrices $A_1,...,A_K$ is given by the formula:

$$\rho^2(X, \Gamma(\bar{a}, L)) = (X - \bar{a}, Z(\wp_{\tilde{\alpha}}^*)(X - \bar{a}))_{tr} = \| X - \bar{a} \|^2 - \sum_{k=1}^{r} (X - \bar{a}, \tilde{U}_k)_{tr}^2$$

where cortege operator $\wp_{\tilde{\alpha}}$ is determined by expression $\wp_{\tilde{\alpha}} = (\tilde{A}_1,...,\tilde{A}_K)$, and $\tilde{U}_i, i = \overline{1,r}$ orthonormal set of eigenmatrices of operator $\wp_{\tilde{\alpha}}^*$.

Proof

Proof is obvious because of $\rho^2(X, \Gamma(\bar{a}, L)) = \rho^2(X - \bar{a}, L)$ and previous theorem.

Parallels between matrix feature vectors and Hu moments

Hu moments are invariant under translation, changes in scale and rotation. Mentioned properties can be effectively used in gesture recognition because it is quite convenient to be able to check if two objects are similar to within rotation or scale etc. The problem is that each gesture has quite strict rules that allow person to show corresponding gesture correctly. In other words it can be acceptable to consider rotation while checking object similarity but only in some interval. However, Hu moments do not allow us to do that and finally they consider objects similar too often so results become not satisfactory. It does not mean that Hu moments are not effective, but it is difficult to use all their advantages for gesture recognition especially on huge set of gestures.

Matrix feature vectors with ellipsoidal or orthogonal compliance distance is used with learning samples. Learning sample is obtained for each gesture and consists of a set of matrices. The matrices correspond to different images of gestures under different environmental conditions. Dictionary of gestures is used in the process of clustering. After converting initial image into the characteristic matrix, this matrix, using one of the compliance distances considered in the article, is checked for closeness to every element from the dictionary. Element that appears to be “the nearest” in the terms of the compliance distance is accepted as a result.

The main goal of both Hu moments and matrix feature vectors is to recognize gestures in different environment: angle of demonstration, distance from hand to recording device etc. However, matrix feature vectors are more stable because they consider not all possible rotations of gestures while comparing them, but only those which are placed in learning set and correspond only to correct demonstrations of gesture. Matrix feature vectors do not have such redundancy as Hu moments have.

Testing and results

Testing of gesture recognition was conducted on the set of dactyls. Specially developed program module formed a training set (base of standards) for every element from the dictionary, using characteristic matrices. Depending on system configuration, ellipsoidal or orthogonal distance was used.

For implementation of programmatic part of task C# was chosen. There is a shell of library “Open CV” for this environment, called Emgu CV. It includes a rich toolkit which allows working with the data flow that is obtained from recording device in real-time. In addition, it contains a number of functions and classes which can be effectively applied to recognition.

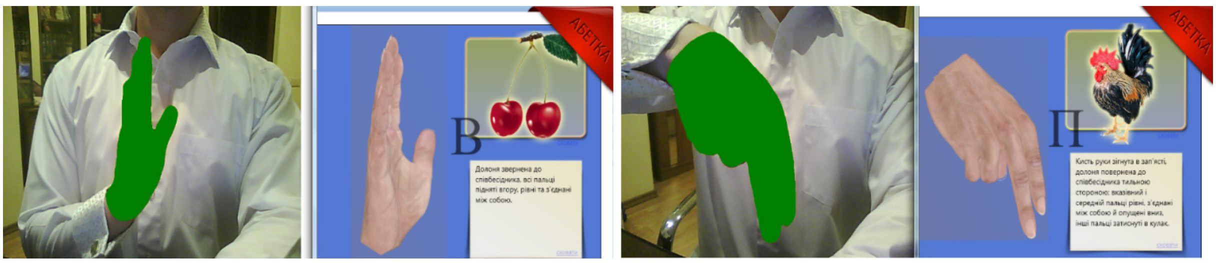


Figure 4. Examples of gesture recognition

Hu moments show the best performance with usage of numerical intervals. Each Hu moment for each gesture was considered separately. After practical part (testing and configuration) appropriate numerical intervals for each Hu moment was found. All in all, we found a set of intervals for each gesture. Size of this set corresponds to amount of Hu moments.

While recognition for initial characteristic matrix we find values of Hu moments and for each gesture check if the values meet corresponding numerical intervals. If all the values get to the intervals then corresponding gesture is marked as “possible solution”. This approach does not guarantee that every time there will be only one “possible solution”, but it is stable enough to be used on practice.

Matrix feature vectors can be used separately or together with Hu moments. Results of testing show that the best variant is to use them on a set of mentioned above “possible solutions” which are found using Hu moments. In this case quality of recognition is really high. The only problem that cannot be solved using suggested approaches appears when gestures look really similar and we need information not only about contour of hand, but about position of each finger.

Figure 5. Illustration of problem with same contour but different finger positions

Figure 5 illustrates the problem. It cannot be solved at this stage, because current solution does not check if fingers are before or behind a palm. Algorithms of a skeletization could be a good choice for this case.

Conclusion

Prospective direction is usage of multilevel clustering, where different technics and algorithms are applied stage-by-stage. Suggested matrix feature vectors and the compliance distances can make a basis of one of such stages. It is a good option to use matrix feature vectors after checking numerical intervals for Hu moments. Although the proposed compliance distances require a subsequent study and optimization, but even on current stage for gesture recognition of tactile language, mathematical results that are illustrated in the applications shows the capacity.

All in all in this article were considered problems of classification tactile language. Note, that matrices are natural representatives of objects in the mentioned task. Development of mathematical apparatus of pseudoinverse for analysis of such objects on the basis of theory of pseudoinverse for matrix Euclidian spaces was proposed.

Acknowledgement

The paper is published with partial support by the project ITHEA XXI of the ITHEA ISS (www.ithea.org) and the ADUIS (www.aduis.com.ua).

Bibliography

1. Golik A.O. Designing the feature vectors: implementation of speech recognition algorithms using parallel computing capabilities. Visn., Ser. Fiz.-Mat. Nayky, Kyiv Univ. im. Tarasa Shevchenka. – 2012. – N 3. – P. 141-150. (in Ukrainian).

2. Donchenko V.S., Golik A.O. Implementation of algorithms of speech recognition using parallel computing capabilities // XVI International Conference “Problems of Decision Making under Uncertainties”(PDMU -2010),– October 4-8, 2010.–Yalta, Ukraine.

3. Donchenko V.S., Golik A.O. Using of parallel computing technology in speech recognition problems // XVII International Conference “Problems of Decision Making under Uncertainties”(PDMU -2011),– May 23-27, 2011.– Skhidnytsia, Ukraine.

4. Donchenko V.S. Euclidian space of vectors and matrices: constructive methods of description of basic structures and their usage // International Journal “Information technologies & Knowledge”.- 2011.- Vol. 5.- Number 3.-P.203-216.

5. Donchenko V., Zinko T., Skotarenko F.’Feature vectors’ in grouping information problem in applied mathematics: vectors and matrices /Problems of Computer Intellectualization.-Editors Vitali Velichko, Aleksey Voloshin O, Krassimir Markov.-Kyiv,Ukrain- Sofia Bulgaria: NASU, V.M.Glushkov Institute of Cybernetics, ITHEA : 2012- p.111-124.

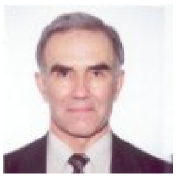
6. M. K. Hu. Visual Pattern Recognition by Moment Invariants. IRE Trans. Info. Theory, vol. IT-8, pp.179–187, 1962.

7. J. Flusser. On the Independence of Rotation Moment Invariants. Pattern Recognition, vol. 33, pp. 1405–1410, 2000.

8. J. Flusser and T. Suk. Rotation Moment Invariants for Recognition of Symmetric Objects. IEEE Trans. Image Proc., vol. 15, pp. 3784–3790, 2006.

9. Donchenko V., Krivonos U., Omardibirova V. Basic structures of Euclidian space: methods of description and usage / Information Models of Knowledge.- Editors: Krassimir Markov, Vitalii Velichko, Oleksy Voloshin - ITHEA.- Kiev, Ukraine – Sofia, Bulgaria. Number 16.– P. 155-169.- ISBN 978-954-16-0048-1.

Authors' Information



Volodymyr Donchenko – *professor, Taras Shevchenko National University of Kyiv, Department of System* Analysis and Decision Making Theory, e-mail: voldon@bigmir.net

Major Fields of Scientific Research: statistical methods in mathematical modeling of objects, Hough transform.

Andrew Golik – *post-graduate student, Taras Shevchenko National University of Kyiv, Department of System Analysis and Decision Making Theory, e-mail: analytical@ukr.net*

Major Fields of Scientific Research: computer vision (especially speech and gesture recognition), artificial intelligence, pattern recognition, algorithms.