Proceedings of the 2014 Federated Conference on
Computer Science and Information Systems pp. 651–658

DOI: 10.15439/2014F365
ACSIS, Vol. 2

Computer Aided Assessment of Linear and Quadratic Function Graphs Using Least-squares Fitting

Sebastian Stoliński, Wojciech Bieniecki
Lodz University of Technology
Institute of Applied Computer Science
al. Politechniki 11, 90-924 Lodz, Poland
Email: {sstolin, wbieniec}@kis.p.lodz.pl

Magdalena Stasiak-Bieniecka
Lodz University of Technology
Department of Electrical Apparatus
ul. Stefanowskiego18/24, 90-924 Lodz, Poland
Email: stasiak@p.lodz.pl

***Abstract*—In this paper an image processing algorithm for automatic evaluation of scanned examination sheets is described. The discussed image contains selected function graphs sketched on a prepared sheets. This type of task is characteristic of final high school exams of natural sciences. Our challenge was to develop an evaluation algorithm, which works with a precision comparable to the teacher. If the image contains the correct solution, the algorithm should husk it from a set of random lines, deletions, amendments, drafts, bearing in mind, that lines were drawn by hand. In addition, the algorithm should calculate scores for partially correct solutions. An essential part of our proposal, which is image segmentation and identification, is based on least-squares fitting combined with 1-NN classification. The proposed solution is flexible and can be extended to other types of tasks such as drawing geometrical figures.**

I. Introduction

ELECTRONIC marking (e-marking), also known as Computer Assisted Assessment (CAA) or Computer Based Assessment (CBA) is relatively a new idea in the field of teaching. Its main advantage is facilitating the laborious process of design, delivery, collection, scoring and analysis of the assessments [10]. Other advantages of CAA are easier schedule and administration of assessments, the immediacy of results, their increased objectivity and security, the possibility of monitoring students and suitability for distance learning [6].

Students also seemed to consider CBA as being more promising, credible, objective, fair, interesting, fun, fast and less difficult or stressful, while they stated that they preferred computerized versus written assessment [7], [5]. In [11] it has been shown, that introduction of CAA allows to keep original accuracy of the exam and increase its reliability and even improve exam quality.

E-marking has been widespread in Great Britain and the USA. The experience gained by Examination Boards like AQA, OCR and EDEXCEL in Great Britain and ETS in the USA suggests that introducing e-marking improves the quality and reliability of the exams.

CAA and e-marking systems described in the literature have been designed for automatic evaluation of the exams carried out at the computer, which means, they use analytical or lexical form immediately.

The prevalence of this form of examination in the case of final examinations in primary and secondary schools can be difficult, due to the need to build IT infrastructure capable to handle a large number of people using the system at the same time.

We anticipate that still the dominant number of examinations will be carried out on the paper sheets and will be checked by humans.

Designing a system that will identify and evaluate the content of the answer sheets based on image processing and understanding algorithms can solve this problem, and in addition will be the value of research in the field of artificial intelligence.

Among the available literature and documentation we have not found any CAA systems that rely on image analysis and could be compared with ours, although there exist possible useful for our problem applications of:

- optical character recognition / intelligent character recognition (OCR/ICR) [18];
- lexicography analysis [13];
- image understanding techniques [14];
- neural networks to OCR/ICR and text identification algorithms [4], [12];
- Hough transform to object identification [9].

Our method of evaluation sketched function graphs relies on

- conversion the sketched shapes to its analytical form (coefficients an equation of a straight line or a parabola)
- merging of graph fragments basing on the evaluated coefficients
- comparison of the sketched graphs to model graphs by comparison of the evaluated coefficients to required values

In contrast to our previous approaches we do not use any reference image (possibly sketched by a teacher).

In practice each image should be segmented into individual primitives before the comparison. In our previous works [15],

[16], [17] we utilized cross-correlation [3] and Generalized Hough Transform (GHT) for this purpose. Unfortunately, the methods, we utilized, did not prove to be flexible and requires major redesign of the algorithm and for assessing new tasks. On the contrary, the least squares method can be used to identify most of the lines described analytically (through equation). The difficulty lies only in the transformation of the figure equation to the form of which the iterative process of fitting is convergent.

II. Description of the Task

The students have been asked to draw two graphs. First one is a function that has two points of discontinuity(eq. 1).

$$f(x) = \begin{cases} -4, & \text{for } x \leq -4 \\ -0.5x + 3, & \text{for } x \in (-4; 4) \\ -x + 9, & \text{for } x \geq 4 \end{cases} \quad (1)$$

Its graph (Fig. 1) consists of three line segments:

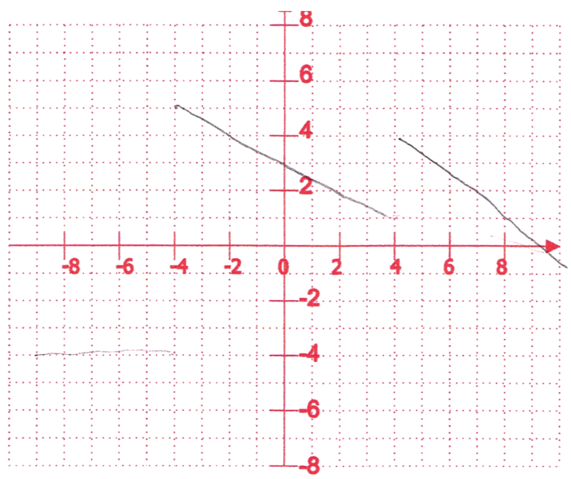

Fig. 1. An example of correctly sketched graph of the linear function

Second one is a parabola given by the formula (eq. 2):

$$f(x) = x^2 - 6 \cdot x + 5 \quad (2)$$

It has zeros in $x_1 = 1$ and $x_2 = 5$ and the minimum in the point $(3, -4)$ (see Fig. 2)

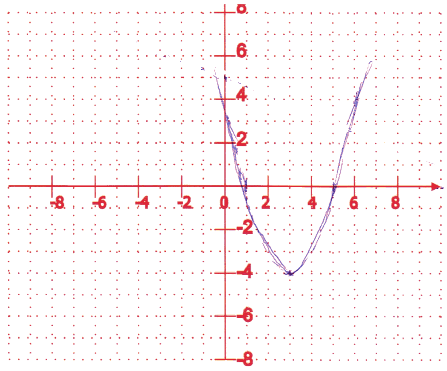

Fig. 2. An example of correctly sketched graph of the parabola

III. Fitting Shapes Using Least-Squares

In the literature one can find a few examples of the use of fitting methods for finding the unknown parameters of geometric figures or function graphs. The publications are related to circles and ellipses [2], spheres, ellipses, hyperbolas, and parabolas [8]. Authors of these studies often adopt a two-phase method: first phase - algebraic fitting, second phase geometrical fitting.

Algebraic fit consists in solving the equation:

$$F(x, z) = \theta \quad (3)$$

where z is a vector of n parameters, x are points in l-dimensional (for example $l = 2$) space. To calculate the parameters for an analytical form of the function using fitting we must create the matrix B for which:

$$[B] = \theta B = \begin{bmatrix} f_1(x_1) & f_2(x_1) & \cdots & f_{k-1}(x_1) & 1 \\ f_1(x_2) & f_2(x_2) & \cdots & f_{k-1}(x_2) & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ f_1(x_m) & f_2(x_m) & \cdots & f_{k-1}(x_m) & 1 \end{bmatrix} \quad (4)$$

Then, for each point $x_p, p = 1 \ldots m$, we have

$$f_1(x_p) \cdot a_1 + f_2(x_p) \cdot a_2 + \ldots + f_{k-1}(x_p) \cdot a_{k-1} + a_k = 0 \quad (5)$$

The algebraic fit usually does lead to the exact solution as it consists in solution of the overdetermined system $(m \gg k)$ and choice the approximate solution to minimize the mean square error. In our approach we have chosen Singular Value Decomposition.

The idea of decomposition of the matrix $[B], (m \times k)$ is creation of a column orthogonal matrix $[U], (m \times m)$, a diagonal matrix $[W](m \times k)$ with zero or non-zero elements and a square, orthogonal matrix $[V](k \times k)$. We obtain the following equation:

$$[B] = [U] \cdot [W] \cdot [V]^T \quad (6)$$

The condition of orthogonality is

$$[U]^T \cdot [U] = [V]^T \cdot [V] = 1 \quad (7)$$

Evaluated matrix V corresponding to smallest singular value of $[B]$ contains a vector of parameters z in its last column.

$$[V] = \begin{bmatrix} v_{11} & \cdots & z_1 \\ v_{21} & \cdots & z_2 \\ \vdots & \ddots & \vdots \\ v_{k1} & \cdots & z_k \end{bmatrix} \quad (8)$$

This solution is taken as the starting point for the iterative process of geometric fit method, which allows a more accurate approximation of the sought parameters. We also control the convergence of the iteration process, and at each step we can estimate the error. This in turn give grounds for considering whether it is possible to fit the selected function to given set of points.

The objective of the geometric fit is to minimize the geometric square error.

$$\epsilon = \sum \|x - x_0\| \tag{9}$$

In this study the Gauss-Newton method is used.

IV. Proposed algorithms

The proposed method of sketched graphs assessment is based on the analysis of all connected components in the image, obtained in the process of preprocessing. In the phase of image preprocessing, all printed content of the examination sheet is erased - only sketched lines remain. However, the process of overprints removal as well as the way in which the student draws a graph causes some defects - the filtered lines may not remain connected, for example, the circle becomes a dozen of arcs. Next the lines are thinned to a single pixel.

Assuming that the exercise was to draw a graph of a linear function, which is a straight line, the algorithm will work as follows:

1) For each connected component, run fitting procedure that finds a, b, c parameters for the equation $a \cdot x + b \cdot y + c = 0$. If the process is not convergent – it means that the component is not a line segment and reject it.
2) Calculated vectors $z = (a, b, c)$ form a feature linear space. Using clustering, find the most similar vectors. This means that the corresponding sets of points belong to one line.
3) for the union of the components obtained in the previous step re-do the fitting to accurately determine the parameters of the line.

This method can be generalized to simultaneously search for several straight lines. Then, the obtained vectors (a, b, c) will be subjected to clustering to indicate several groups of similar vectors.

In case the student task is to draw a number of different geometric shapes, we will try to adjust the parameters of different functions to each of them, looking for the best fit.

A. Fitting a straight line

In the first phase, the algorithm will minimize the algebraic error. Assume that a simple algebraic representation a straight line in the plane is given by

$$\begin{aligned} &F(x) = A^T \cdot x + c = 0, \\ &A = (a, b) \in \mathbb{R}^2, \\ &x \in \mathbb{R}^2, \\ &c \in \mathbb{R} \end{aligned} \tag{10}$$

The aim is to find values of a, b, c for given points x. Substituting the coordinates of the points in the above equation we obtain the system of equations. $[B] \cdot z = 0$, for the parameters $z = (a, b, c)$, where B is in the form:

$$[B] = \begin{bmatrix} x_{11} & x_{12} & 1 \\ x_{21} & x_{22} & 1 \\ \vdots & \vdots & \vdots \\ x_{m1} & x_{m2} & 1 \end{bmatrix} \tag{11}$$

Assuming, that $m > 3$, B is a rectangular, the system is overdetermined and probably inconsistent. The solution is approximated with Singular Value Decomposition.

Denote the obtained solution by $z = (a_0, b_0, c_0)$.

In the case of good fit such a solution is sufficient. However, if there are points x lying far from the approximated line, the bias arises, and therefore a second phase of the algorithm – geometric fit – must be launched.

Since there are many combinations of (a_0, b_0, c_0) corresponding to one line, one of the co-ordinates should be eliminated. Choose $z_M = \max(|a_0|, |b_0|, |c_0|)$ and assume, that $z = (1, b_0/z_M, c_0/z_M)$ or $z = (a_0/z_M, 1, c_0/z_M)$ or $z = (a_0/z_M, b_0/z_M, 1)$

The Gauss-Newton method involves the iteration which consists of two operations:

- solution of the system $-[J] \cdot h = f$ with the unknown vector h;
- correction of the solution $z = z + h$.

In the above system f is the objective function. It is a vector of distances of each point to the fitted line. J is the Jacobian - contains derivatives of the coordinates of the vector z (sought parameters). Formally:

$$\begin{aligned} &f = (f_1, f_2, \dots, f_m); \quad f_i = \tfrac{|ax_{1i} + bx_{2i} + c|}{\sqrt{a^2+b^2}}; \\ &J = \begin{bmatrix} \frac{\partial f_1}{\partial a} & \frac{\partial f_1}{\partial b} & \frac{\partial f_1}{\partial c} \\ \frac{\partial f_2}{\partial a} & \frac{\partial f_2}{\partial b} & \frac{\partial f_2}{\partial c} \\ \vdots & \vdots & \vdots \\ \frac{\partial f_m}{\partial a} & \frac{\partial f_m}{\partial b} & \frac{\partial f_m}{\partial c} \end{bmatrix}; \\ &J_{i1} = \frac{\operatorname{sgn}(ax_{1i}+bx_{2i}+c) \cdot x_{2i} - |ax_{1i}+bx_{2i}+c| \cdot \frac{a}{\sqrt{a^2+b^2}}}{a^2+b^2} \\ &J_{i2} = \frac{\operatorname{sgn}(ax_{1i}+bx_{2i}+c) \cdot x_{2i} - |ax_{1i}+bx_{2i}+c| \cdot \frac{b}{\sqrt{a^2+b^2}}}{a^2+b^2} \\ &J_{i1} = \frac{\operatorname{sgn}(ax_{1i}+bx_{2i}+c)}{\sqrt{a^2+b^2}} \end{aligned} \tag{12}$$

The condition of convergence is calculated by the relative difference between the current and the previous solution.

$$\Delta = \frac{\|h\|_\infty}{\|z\|_\infty} \tag{13}$$

B. Fitting a parabola

In our discussion we will consider only the parabola which symmetry axis is parallel to the y-axis in the coordinate system. Accordingly, the parabola is defined by the algebraic equation:

$$F(x) = a \cdot x_1^2 + b \cdot x_1 + c \cdot x_2 + d = 0 \tag{14}$$

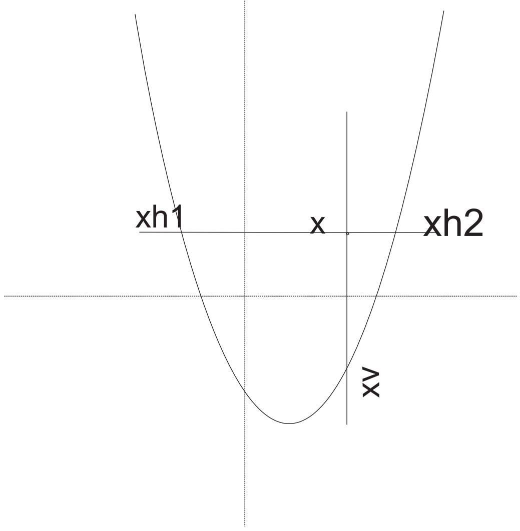

Fig. 3. Approximation of the distance of the point from the parabola

Denote the searched value $z = (a, b, c, d)$. To find the algebraic fitting for the equation denote:

$$[B] = \begin{bmatrix} x_{11}^2 & x_{11} & x_{12} & 1 \\ x_{21}^2 & x_{21} & x_{22} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_{m1}^2 & x_{m1} & x_{m2} & 1 \end{bmatrix} \tag{15}$$

Just as in the case of a straight line, the system of equations is solved by SVD giving the value of $z_0 = (a_0, b_0, c_0, d_0)$. Then the geometric fit is carried out.

Similarly, as in the case of a straight line, in order to avoid ambiguity (and, consequently, the divergence of the process) we eliminate one of the parameters. Given the assumed position of the parabola it will be a parameter c.

$$F(x) = a/c_0 \cdot x_1^2 + b/c_0 \cdot x_1 + x_2 + d/c_0 = 0 \tag{16}$$

Iterative process will be conducted the same way as in the case of a straight line. However, due to a difficulty in determining the distance from the point to the parabola, some simplification will be used.

To calculate the distance of a point x to the parabola together with its partial derivatives, we derive a straight line from this point, parallel to the x-axis. It may cross the parabola at points X_{H1} and X_{H2}. We also derive a line from x, parallel to the y-axis, which always intersects the parabola (the point X_V). The points of intersection are calculated straight out of the equation of the parabola. Denote:

$$\begin{array}{l} f_x = \min(\rho(x, x_{h1}); \rho(x, x_{h2})) \\ f_y = \rho(x, x_v) \end{array} \tag{17}$$

If x_{h1} and x_{h2} do not exist, assume that $f_x = f_y$.

Assuming that x_{H1} is a closer point, we calculate the partial derivatives of its distance.

$$\begin{array}{l} \frac{\partial f_x}{\partial a} = \operatorname{sgn}(x_1 - x_{h1,1}) \cdot b \cdot (b + \sqrt{\Delta} + 2a(x_2 + c)/\sqrt{\Delta})/2a^2 \\ \frac{\partial f_x}{\partial b} = \operatorname{sgn}(x_1 - x_{h1,1}) \cdot (-1 - b/\sqrt{\Delta})/2a \\ \frac{\partial f_x}{\partial c} = \operatorname{sgn}(x_1 - x_{h1,1}) \cdot \sqrt{\Delta} \\ \frac{\partial f_y}{\partial a} = \operatorname{sgn}(x_2 - x_{v,2}) \cdot x_1^2 \\ \frac{\partial f_y}{\partial b} = \operatorname{sgn}(x_2 - x_{v,2}) \cdot x_1 \\ \frac{\partial f_y}{\partial c} = \operatorname{sgn}(x_2 - x_{v,2}) \end{array} \tag{18}$$

As an approximate distance of the point to the parabola we use the geometric average of calculated values f_x, f_y.

$$sf = f_x^2 + f_y^2 \qquad f = \frac{f_x \cdot f_y}{\sqrt{sf}} \tag{19}$$

Then, the Jacobian is defined by equations.

$$\begin{array}{l} J_1 = \left(\left(\frac{\partial f_x}{\partial a} f_y + \frac{\partial f_y}{\partial a} f_x \right) \sqrt{sf} - f \left(\frac{\partial f_x}{\partial a} f_x + \frac{\partial f_y}{\partial a} f_y \right) \right) /sf \\ J_2 = \left(\left(\frac{\partial f_x}{\partial b} f_y + \frac{\partial f_y}{\partial b} f_x \right) \sqrt{sf} - f \left(\frac{\partial f_x}{\partial b} f_x + \frac{\partial f_y}{\partial b} f_y \right) \right) /sf \\ J_3 = \left(\left(\frac{\partial f_x}{\partial c} f_y + \frac{\partial f_y}{\partial c} f_x \right) \sqrt{sf} - f \left(\frac{\partial f_x}{\partial c} f_x + \frac{\partial f_y}{\partial c} f_y \right) \right) /sf \end{array} \tag{20}$$

V. The main algorithm for image processing

Due to the fact that each image may have a different content and may include various types of function graph we decided to identify the graph using least-squares method for each line segment found in the image.

The preprocessing phase includes separation of the drawing from the rest of scanned examination sheet. Our method described in [16] has been replaced by more efficient color discrimination. For this purpose the coordinate system had to be printed red, while students use blue or black pen.

The color filtration condition is presented by the formula

for all pixel p IN I_l **do**
  if $|red(p) - green(p)| > 35 AND |red(p) - blue(p)| > 20$ **then**
    $p = (255, 255, 255)$
  end if
end for

Next steps of the preprocessing algorithm:

- image binarization using Otsu method [1];
- using morphological filters thin the lines, remove isolated points;

$$interval = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{21}$$

- using hit-miss transform detect and remove all crossings – trench the crossing lines;

$$interval = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{22}$$

- segment the image – label all connected components. We obtain I_l image containing $1 \ldots l$ components.

The following sections present our methods of analyzing and assess individually: discontinuous linear function and quadratic function.

A. Analyzing straight lines

The main part of the image processing is summarized in the following algorithms

Algorithm 1 Processing of connected components of I_l

$count \leftarrow 0$; $tolerance \leftarrow 1e-5$;
while $d > tolerance$ AND $count < 10$ **do**
  for all $line$ IN I_l **do**
    if $size(line) > 20$ **then**
      $(a_0, b_0, c_0) \leftarrow$ AlgebraicFitLine $(line)$
      $W_i \leftarrow (a, b, c, x_{\min}, x_{\max}, y_{\min}, y_{\max})_i \leftarrow$ GeometricFitLine $(line, a_0, b_0, c_0)$
    end if
  end for
  $d \leftarrow \max(res)$
  $count \leftarrow count + 1$
  $lines \leftarrow$Merge$(lines)$
end while

Here is the explanation of the Alg. 1. Repeat the operations until the error of fit is greater than expected: For each connected component, greater than 20 pixels do the operations: (if the number of iterations reaches 10, the procedure breaks regardless the obtained result)

1) algebraic fit (see sec. IV-A). A result is a vector (a_0, b_0, c_0)
2) geometric fit (see sec. IV-A). A result is a vector (a, b, c) and the convergence error res. The procedure of geometric fit is iterative and the iteration breaks if the expected convergence is reached ($tol < 1e-3$) or the number of iterations exceeds the established number ($maxiter > 10$).
3) find two nearest connected components and if the distance and merge them (assign the same label)

From the set of recognized lines we choose these, that lie nearest to the model lines. This is required, when the graph to be drawn consists of a few line segments (the function is not continuous). Model line parameters are obtained from the analytical form of the function. The comparison is carried out using 1-Nearest Neighbor statistical classifier, with the feature space identical to W_i vector in Alg. 1.

B. Obtaining a score

Each of line segments of the discontinuous function which is to be drawn is assessed independently, so if there are three segments of the graph, the maximum mark is 3. We take into account A, B, C coefficients of the linear function and the minimum and maximum x coordinate of the found line.

C. Analyzing a parabola

The initial phase of the algorithm is similar to Alg. 1 for processing straight lines

Algorithm 2 Processing of connected components of I_l

$count \leftarrow 0$; $tolerance \leftarrow 1e-1$;
while $d > tolerance$ AND $count < 10$ **do**
  for all $line$ IN I_l **do**
    if $size(line) > 20$ **then**
      $(a_0, b_0, c_0, d_0) \leftarrow$ AlgebraicFitParabola $(line)$
      $W_i \leftarrow (a, b, c, x_{\min}, x_{\max}, y_{\min}, y_{\max})_i \leftarrow$ GeometricFit $(line, a_0, b_0, c_0, d_0)$
    end if
  end for
  $d \leftarrow \max(res)$
  $count \leftarrow count + 1$
  $lines \leftarrow$Merge$(lines)$
end while

The main loop is repeated until we reach a required tolerance but no more than 10 times. As for straight lines, the algebraic fit followed by geometric fit are calculated. Again, as for straight lines, we merge the neighboring curves. The merge is done conditionally. For each pair the fitting is performed as for one set of points. If the convergence does not increase more than three times, the curves may be merged. After completion the process of fitting and merging the curves we obtain one or more curves that are possibly parabolas.

For the process of the assessment we take into account:

- (a, b, c) coefficients for the parabola equation $a \cdot x^2 + b \cdot x + c = 0$
- $(x_{\min}, x_{\max})$ - a position of the curve in the coordinate set.
- $(y_{\min}, y_{\max})$ - minimum or maximum value of the drawn function
- a count of pixels, that are not assigned as parabolas (these are possibly amendments)

If for the examined graph true are the statements:

1) values $(x_{\min}, x_{\max})$ and ($y_{\min}$ or $y_{\max}$) fall within a specified range
2) two of (a, b, c) parameters fall within a specified range
3) the count of amendment pixels is less than a specified threshold

the student receives 1 point. Moreover if all of (a, b, c) parameters fall within a specified range, the student receives a maximum score - 2 points.

If the count of amendment pixels exceeds a specified threshold, the graph is assigned as unrecognized.

VI. Teaching the algorithms

The algorithm for graphs classification runs in a supervised manner (with teaching). For each new type of the image (modified print-out, different task, different scanner) the step of teaching must be repeated. The teaching phase consists of evaluation acceptable ranges in the feature space. Some of them are calculated from the formula of the task:

- a, b, c parameters (both lines and parabola)
- x_{min} and x_{max} values

- y_{min} or x_{max} value (only for a parabola)

but their tolerances must be evaluated experimentally. Other parameters are:

- a threshold for detection of amendments
- a threshold for minimal length of a line (only for lines)

The teaching is carried out by comparing the proper scores (given by a teacher) for several test images containing graph sketches with the scores calculated by algorithms.

The aim of the tune-up is to minimize the overall error (number of different scores)

For the experiments 57 sheets with Task 1 and 72 sheets with task 2 have been used. All the sheets have been assessed by teachers for comparison. The examination sheets have been scanned in color mode with resolution 300 DPI. With this resolution each image containing extracted coordinate set with sketches has an area about 1 Megapixel.

Table I contains results of manual assessment of Task 1 for 11 exemplary works and the parameters obtained by an algorithm.

TABLE I
TASK 1: 11 EXEMPLARY SHEETS - ASSESSED BY A TEACHER

Sample	score for a segment 1	2	3	Total Score	Notes
1	1	1	1	3	
2	0	0	0	0	
3	0	0	0	0	
4	1	1	1	3	additional lines
5	1	1	1	3	
6	0	0	1	1	
15	0	0	0	0	
18	1	0	1	2	strike-throughs
19	1	1	1	3	
20	1	1	1	3	strike-throughs
21	1	1	1	3	strike-throughs

To the process of algorithm teaching 30 of 57 sheets have been randomly drawn (summarized in Table II).

TABLE II
TASK 1: THE PROCESS OF TEACHING

Property	Value
samples – in total	57
samples in a training	30
samples in a testing set	27
Segment 1 A	0 ± 0.0009
Segment 1 B	0.0015 ± 0.0007
Segment 1 C	1 ± 0.05
Segment 1 $x_{\min}$	45 ± 45
Segment 1 $x_{\max}$	340 ± 25
Segment 2 A	0.018 ± 0.0179
Segment 2 B	0.045 ± 0.043
Segment 2 C	1 ± 0.05
Segment 2 $x_{\min}$	340 ± 35
Segment 2 $x_{\max}$	770 ± 35
Segment 3 A	-0.0017 ± 0.0005
Segment 3 B	-0.0017 ± 0.0015
Segment 3 C	1 ± 0.05
Segment 3 $x_{\min}$	750 ± 25
Segment 3 $x_{\max}$	1050 ± 50

Note, that the values of parameters presented in Table II are expressed in pixels rather than units.

Tables III and IV present corresponding data for Task 2

TABLE III
TASK 2: 11 EXEMPLARY SHEETS - ASSESSED BY A TEACHER

Sample	score	Notes
10	1	
11	2	
12	2	
13	2	
14	1	
15	0	
16	1	
17	0	
18	0	additional objects
19	0	
20	1	

TABLE IV
TASK 2: THE PROCESS OF TEACHING

Property	Value
total samples	72
training set cardinality	36
testing set cardinality	36
Parameter A	-0.02 ± 0.015
Parameter B	28 ± 4
Parameter C	-10000 ± 5500
Parameter $x_{\min}$	530 ± 120
Parameter $x_{\max}$	880 ± 150
Parameter $maxpix$	< 600

VII. THE RESULTS OF THE EXPERIMENT

In Table V the best results (for parameters presented in Tables II and IV) for training sets have been summarized.

TABLE V
THE OF RESULTS FOR TRAINING SETS COMPARED TO TEACHER SCORES

Item	Test 1	Test 2
Samples	36	36
Unrecognized	3	1
Underestimated score	1	0
Overestimated score	2	1
Compliant score	30	34
Recognized compliant ratio	83%	94%

In Figs 4 – 8 exemplary correct and incorrect results are presented.

Table V summarizes the results obtained for testing sets.

TABLE VI
THE OF RESULTS FOR TESTING SETS COMPARED TO TEACHER SCORES

Item	Test 1	Test 2
Samples	36	36
Unrecognized	6	0
Underestimated score	2	1
Overestimated score	1	5
Compliant score	27	30
Recognized compliant ratio	75%	83%

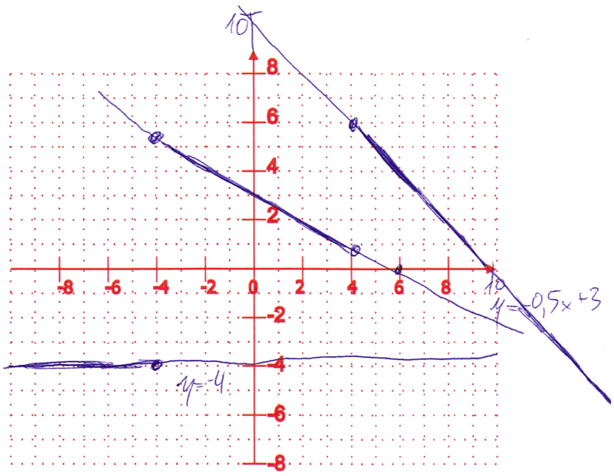

Fig. 4. Task 1, sample 4. A correct solution - but the algorithm did not recognize the sketches

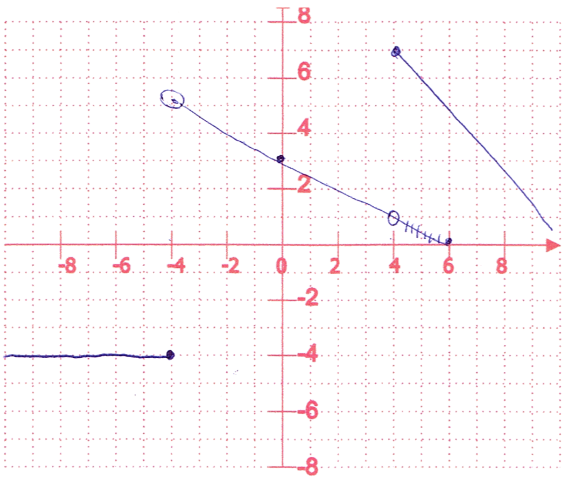

Fig. 5. Task 1, sample 29. A correct solution - underestimated by the algorithm. The central line has been disqualified.

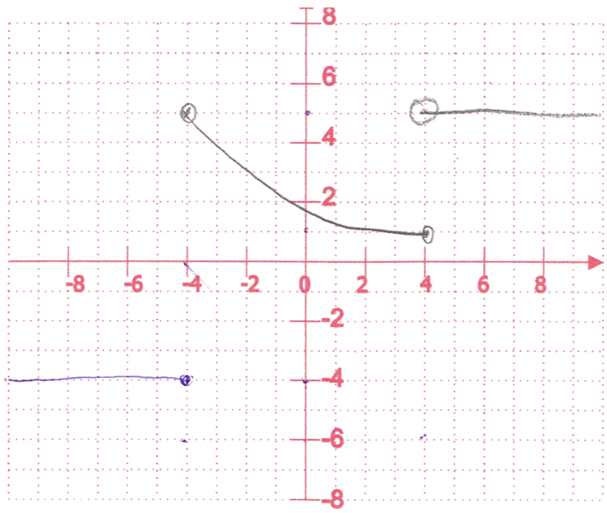

Fig. 6. Task 1, sample 32. An incorrect solution - the central line is not straight. The algorithm qualified this line as correct.

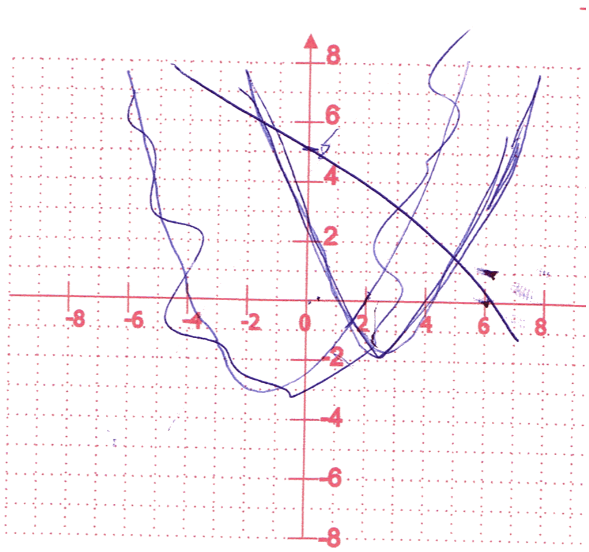

Fig. 7. Task 2, sample 18. Unrecognized sketches

VIII. Conclusions

According to this report (Table VI) the algorithm works properly for the case of the tested task. The errors occurred for the cases when the solutions contained amendments, strike-throughs. Tasks which were not recognized, contained additional objects or the drawings were done careless. Parameters of the algorithm were chosen so as to minimize the amount of erroneous assessments among the training set, even at the cost of increase of the number of unresolved tasks.

A general drawback of this approach is a necessity to train the algorithm before an assessment of a new task, but some of the parameters may be read from the analytical form of the function. Other parameters, which are tolerance, can be expressed as a percentage of the size of the unit.

Our future research will include the detection and assessment of graphs of trigonometric, exponential and rational functions. We'll try to extract multiple types of function graphs from one sketch (the task may include drawing a graphical solution of the set of inequalities)

Furthermore the algorithm of identification acceptable and redundant objects will be improved.

References

[1] Otsu, Nobuyuki: A Threshold Selection Method from Gray-level Histograms. IEEE Transactions on Systems, Man and Cybernetics **9**(1), 62–66 (1979). DOI 10.1109/TSMC.1979.4310076

[2] Gander, Walter and Golub, GeneH. and Strebel, Rolf: Least-squares fitting of circles and ellipses. BIT Numerical Mathematics **34**(4), 558–578 (1994). DOI 10.1007/BF01934268. URL http://dx.doi.org/10.1007/BF01934268

[3] J P Lewis: Fast normalized cross-correlation (1995)

[4] Charalambos Strouthopoulos and Nikos Papamarkos: Text identification for document image analysis using a neural network. Image Vision Comput. **16**(12–13), 879–896 (1998). DOI 10.1016/S0262-8856(98)00055-9

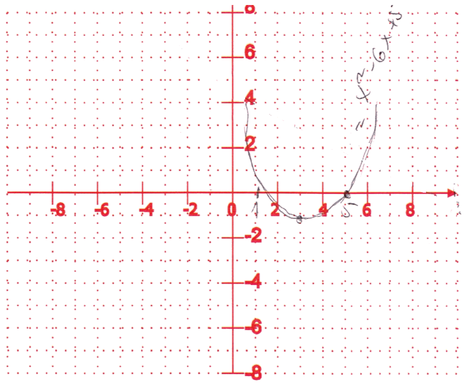

Fig. 8. Task 2, sample 39. Wrong solution (teacher scored 0 point), algorithm scored 1 point (partially correct).

[5] K Sambell and A Sambell and G Sexton: 19, , pp.179–192. Kogan Page, London (1999)

[6] M. Thelwall: Computer-based assessment: a versatile educational tool. Computers and Education **34**(1), 37–49 (2000). DOI 10.1016/S0360-1315(99)00037-8

[7] A C Croft and M Danson and B R Dawson and J P Ward: Experiences of using computer assisted assessment in engineering mathematics. Computers and Education **37**(1), 53–66 (2001). DOI 10.1016/S0360-1315(01)00034-3

[8] Sung Joon Ahn and Wolfgang Rauh and Hans-Jürgen Warnecke: Least-squares orthogonal distances fitting of circle, sphere, ellipse, hyperbola, and parabola. Pattern Recognition **34**(12), 2283 – 2303 (2001). DOI http://dx.doi.org/10.1016/S0031-3203(00)00152-7. URL http://www.sciencedirect.com/science/article/pii/S0031320300001527

[9] Sewisy, Adel A.: Graphical techniques for detecting lines with the hough transform. Int. J. Comput. Math. **79**(1), 49–64 (2002). DOI 10.1080/00207160211911

[10] Gavin Sim and Phil Holifield and Martin Brown: Implementation of computer assisted assessment: lessons from the literature. Research in Learning Technology **12**(3) (2004). DOI 10.1080/0968776042000259546

[11] D Fowles C Adams: How does assessment differ when e-marking replaces paper-based marking? 31st International Association for Educational Assessment Conference, Abuja, Nigeria, 4-9 September 2005 (2005)

[12] Shashank Araokar: Visual character recognition using artificial neural networks. CoRR **abs/cs/0505016** (2005)

[13] Prusa, D. and Hlavac, V.: Ninth International Conference on Document Analysis and Recognition, 2007. ICDAR 2007. pp., 849 –853 (2007). DOI 10.1109/ICDAR.2007.4377035

[14] Ryszard Tadeusiewicz and Marek R. Ogiela and Piotr S. Szczepaniak: Notes on a linguistic description as the basis for automatic image understanding. Applied Mathematics and Computer Science **19**(1), 143–150 (2009). DOI 10.1.1.390.8222

[15] Sebastian Stoliński and Wojciech Bieniecki and Jacek Stańdo: Automatic detection and evaluation of the spline function plot. Automatyka **14/3/1**, 879–896 (2010). DOI http://dx.doi.org/10.7494/automat

[16] Wojciech Bieniecki and Jacek Stańdo and Sebastian Stoliński: Automatic evaluation of examination tasks in the form of function plot. pp., 140–143. Polytechnic National University (2010)

[17] Sebastian Stoliński and Wojciech Bieniecki: The algorithms for automatic evaluation of selected examination tasks from the geometry. Automatyka **15/3**, 551–560 (2011). DOI http://dx.doi.org/10.7494/automat

[18] Sebastian Stoliński and Wojciech Bieniecki: Application of ocr systems to preprocessing and digitalization of paper documents. pp., 102–111. WULS Press, Warszawa (2011)