

The Computer-Aided Detection of Inferior Printing Quality and Errors

Abstract—This paper describes a new approach which detects inferior printing quality and errors by using a regular PC and document scanner. Our method relies on the comparison of an inspected document with its referential version. It firstly registers the images of the two documents, and then detects any discrepancies between the aligned pixels or regions. Iterations of the two-step registrations with interim interpolations introduce a sort of elastic image correction.

We confirmed experimentally that the error detection rate for those documents with simpler structures, mostly pictures, was about 95%, whereas with more complex documents containing a lot of text this figure is about 90%.

I. INTRODUCTION

In many cases we are still obliged to use printed matter, even the use of e-documents is on the rise. Characteristic examples are user manuals, directions for use, product labels and similar, which all abound regarding everyday commercial goods, and mostly cannot be replaced by digital, electronic solutions. Therefore, every industrial company using mass-production is constantly purchasing both new and new series, printed matter, usually from a few specific printing houses. Figures from the cosmetic industry, for example, are impressive: a multinational company utilize over 350 million pieces of printed matter per year. It is quite obvious that undetected inferior printing quality before application onto products, may cause considerable revenue losses.

There are at least two reasons why humans during a company's production inspection, may have difficulties in detecting inferior printing quality: firstly, they probably do not understand all the text being inspected, e.g. Chinese characters, and secondly, their concentration and perception capability span reduces rapidly during routine activities, such as comparing two images, one referential and the other inspected. However, the necessity for quality inspection becomes even more important for those commercial products whose labels are regulated by law, such as warnings of possible allergenic ingredients. Inspection must be reliable, speedy and extensive, and should be automated. All erroneous or just badly printed labels, user manuals or imprinted packaging must be recognized and discarded.

Many commercially available devices have been developed in order to handle the inspection of imprints automatically. They can be classified by the way they acquire images. The first type of device acquires images by applying video cameras, and their usage is mainly in industrial mass-production environments [4], [6]. Their job is to scrutinize the production processes, such as bottling, packing, etc. The second type relies on scanning devices [5] and is primarily used when dealing with printed matter. Prac-

tically all these devices require special, controlled environments for their operation. This means that images can only be acquired under certain illuminated conditions, always in the same position [6], and perspective. They also require the inspected printed matter to contain additional control markers in order to facilitate comparison with referential specimens.

Encouraged by the collaboration of an industrial partner, we decided to develop our own program package, which should overcome the limitations of present devices for the automated inspection of printed matter. In particular, we wanted to abolish the additional printing of control markers and avoid the necessity for controlling the acquisition environments. The methods applied and described in this paper make use of advanced algorithms for image registration and are based solely on a content comparison of the inspected images. No extra, expensive and specialized equipment is required, because our programs run on regular PCs and utilize any document scanner.

The described printing error detection procedure registers an inspected image to the referential one and then compares them for possible discrepancies in a pixel-by-pixel manner. Section II reveals the theoretical background for our computer implementation. Section III describes briefly the graphical user interface along with its operation and functions. Section IV is devoted to experiments with artificially introduced printing errors and also with real examples of inferior printing quality. We assessed the sensitivity and positive predictivity of our algorithms in noisy environments with different signal-to-noise ratios (SNRs). The last section discusses the obtained results and concludes the paper.

II. AUTOMATED PRINTING ERROR DETECTION

Our prototype solution for automated printing error detection assumes the inspected documents are scanned, so that their internal computer representation is by bit images. By firstly scanning a referential document, and then the inspected one, computer-aided printing error detection can be accomplished by a comparison of the obtained images (Fig. 1). This is the reason why, in the reminder of our paper, we talk about images and do not discriminate whether they originated as product labels, user manuals, or package imprints.

Printing quality inspection must be reliable and suitably accurate, regardless of the type of document inspected. This depends on the scanning resolution, i.e. the image resolution, on the one hand and on the image comparison procedure on the other. Based on industrial experience [8], printing discrepancies smaller than 0.2 mm in diameter are undetected by the naked eye. It is, therefore, sufficient that

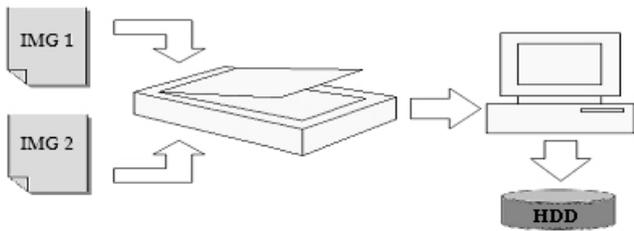


Fig. 1. Recognition system using a scanner to acquire a referential and inspected image and a PC for possible inferior printing quality detection.

automated error detection draws the inspector's attention only to those regions surpassing the following diameter:

$$d \geq \frac{L}{25.4} 0.2 \text{ [dots]} \quad (1)$$

where L stands for image resolution in dots per inch (dpi). Dealing with 300 dpi, the smallest erroneous region looked for would then be 2 to 3 pixels in diameter.

After scanning two documents, a referential one and an inspected one, the obtained images' positions and orientation are expressed within the scanner coordinate system. It is clear, that without any special precautions, the two images would appear translated and rotated towards each other. In order to compare their content for possible printing discrepancies, they have firstly to be totally aligned.

We applied image registration to compensate for rotation and translation. From among several possible registration techniques, we decided on affine transformations [2], primarily because of favourable reports on its flexible and successful implementation regarding medical images [1]. However, this kind of registration is prone to missing the global registration optimum, if the images to be aligned are very complex and considerably displaced. Therefore, we propose a two-step registration procedure, where a coarser alignment of the images precedes the affine transformation.

A. Coarse image registration

Our approach applies alignment of the images' centres of gravity. Firstly, we binarize each image based on the grey-level histogram they produce [2]. Our goal is to separate the image background from the foreground's contents. We assume that the grey-levels representing the background pixels appear most frequently. So we start our binarization by a global maximum search of the histogram. From this point, we descent down both sides until the left and right minimums are reached. The two minimums determine the grey-level interval supposedly belonging to the image background. However, real histograms are not smooth and unimodal. This may cause the background search to terminate prematurely due to some noise disturbance. To minimize the uncertainty, we continue searching for the histogram minimums over six additional iterations. Each successive iteration tolerates just one discontinuity more, in the descent from the global maximum, i.e. the first iteration one discontinuity, the second two, etc.

The described procedure generates six pairs of grey-level thresholds for the presumed background both in the referential and inspected images. These thresholds are then used in the following step for image binarization. The binarized images are then calculated for their centres of gravity, as shown by Eqs. (2):

$$\begin{aligned} t_i &= \sum_{x=1}^M \sum_{y=1}^N b(x, y) \\ x_{t_i} &= \frac{\sum_{x=1}^M \sum_{y=1}^N b(x, y) \cdot x}{t_i} \\ y_{t_i} &= \frac{\sum_{x=1}^M \sum_{y=1}^N b(x, y) \cdot y}{t_i} \end{aligned} \quad (2)$$

where t_i stands for the number of used pixels, $b(x, y)$ means an $M \times N$ binarized image, while x and y determine the pixel coordinates.

The corresponding pairs of binarized referential and inspected images are then aligned with their centres, in order to determine the mean square error (MSE) between them. For the i -th pair of images, it is calculated as $D(i)$:

$$D(i) = \sqrt{\sum_{x=1}^M \sum_{y=1}^N [p_i(x, y, t) - p_i(x, y, t-1)]^2} \quad (3)$$

where $p_i(x, y, t)$ and $p_i(x, y, t-1)$ represent the pixel values of the i -th pair of images, the referential denoted by argument t , the inspected by $t-1$.

After assessing all six pairs, the one with the minimum $D(i)$ is selected for further processing. We have to be aware that image alignment according to their centres of gravity partially eliminates the translation differences, while the rotation matching still has to be achieved. We decided to rotate the inspected image over a range from -5° to 5° , utilizing MSE, to determine the best registered location. To speed up the process, we started in 5° steps and continue with 1° refinement only from the best 5° position, limited to a range of -4° to 4° . Interpolation [3] is used when image rotation is applied in order to overcome the quantisation limitations.

Finally, after the rotation match is obtained, additional translation compensation is applied in a locally limited way. The inspected image is firstly translated in steps of two pixels, and then in steps of one pixel, until the optimum match is discovered with respect to MSE.

B. Fine image registration

Coarse image alignment cannot guarantee such precise image matching, that the inferior printing quality discrepancies can be detected reliably. Furthermore refinement must be taken into account. We implemented image registration [1] based on affine transformation iterations and intermediate image interpolation. This approach proved an extra elastic image correction capability.

Summarize the approach from [1] and apply the image pixels notation from the previous subsection: let $p(x, y, t)$

and $p(x, y, t - 1)$ stand for the pixels of referential and inspected images, respectively. An affine transformation can be used for a rigid registration of one image to the other:

$$p(x, y, t) = p(m_1x + m_2y + m_5, m_3x + m_4y + m_6, t - 1) \quad (4)$$

where parameters from m_1 to m_6 denote the affine transformation parameters.

Using Eq. (4), we can estimate the registration square error between the inspected and referential image as follows:

$$E(\vec{m}) = \sum_{x=1}^M \sum_{y=1}^N [p(x, y, t) - p(m_1x + m_2y + m_5, m_3x + m_4y + m_6, t - 1)]^2 \quad (5)$$

The estimated error in (5) may be minimized by properly defined affine parameters. To calculate them, they must appear in explicit form, which is not the case in Eq. (5). Therefore, the first-order Taylor series is applied, approximating the error function as follows:

$$E(\vec{m}) \approx \sum_{x=1}^M \sum_{y=1}^N \{p(x, y, t) - [p(x, y, t) + (m_1x + m_2y + m_5 - x)p_x(x, y, t) + (m_3x + m_4y + m_6 - y)p_y(x, y, t) - p_t(x, y, t)]\}^2 \quad (6)$$

where p_x , p_y and p_t designate the spatial derivatives according to x , y and t , respectively. A considerable simplification of Eq. (6) is possible if the following expressions are introduced: $\vec{c} = [xp_x, yp_x, xp_y, yp_y, p_x, p_y]^T$ and $k = p_t + xp_x + yp_y$. The approximation error function (6) substitutes into:

$$E(\vec{m}) \approx \sum_{x=1}^M \sum_{y=1}^N (k - \vec{c}^T \vec{m})^2 \quad (7)$$

By taking the partial derivatives to the parameters m of (7) and making it equal to 0, the final values for \vec{m} yield:

$$\vec{m} = \left(\sum_{x=1}^M \sum_{y=1}^N \vec{c} \vec{c}^T \right)^{-1} \left(\sum_{x=1}^M \sum_{y=1}^N \vec{c} k \right) \quad (8)$$

The registration parameters obtained by Eq. (8) are merely one step when converging towards the best image fit. It should be understood that the compared images may not have exactly the same contents and size. Moreover, the mathematical approximations applied cause the performed registration to reach only the nearest local optimum, in the MSE sense. Search for the global optimum must continue over several iterations, until the MSE decreases. After each iteration, an interim interpolation is applied to both the referential and inspected images, a half of the calculated rotational and translational adjustment to each image. Such a solution keeps the interpolation-caused degradation at approximately the same in both images, and

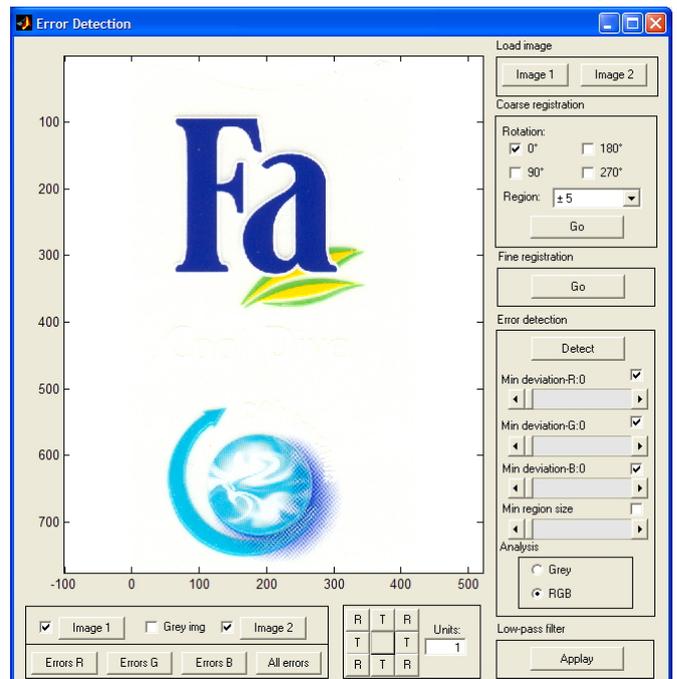


Fig. 4. Graphical user interface: the central working area always shows a selected image, the buttons on the right-hand side turn on the steps for automated image registration and error detection, while the buttons at the interface's bottom control the manual image registration and supervision.

prevents the generation of artificial discrepancies which are too obvious.

Fine image registration is conducted using grey-level images to speed up the process. In the end, cumulative affine parameters are used to also align the original colour images.

III. COMPUTER IMPLEMENTATION

A computer prototype application was implemented according to the proposed solutions from Section II. Fig. 4 depicts the layout of the implemented graphical user interface.

The central working area is designed for displaying the overlaid referential and inspected images over all steps, from their loading and registration, to their printing error detection. It can also depict only a single image if so selected by buttons at the bottom of the user interface.

The buttons on the right-hand side of the screen turn on the steps for automated image registration and error detection. Coarse registration can be done with four initial rotations of 0°, 90°, 180°, or 270°, which are followed by rotations over a preselected interval, as described in Section II.

The step for fine registration can be done either on grey-level images or colour images, according to selection in the lower right part of the user interface. The upper three sliders above set the thresholds for the individual colour planes (red, green, blue), respectively, while the fourth one controls the threshold for the minimum erroneous regions to be displayed.

The right-hand side's bottom button turns on low-pass

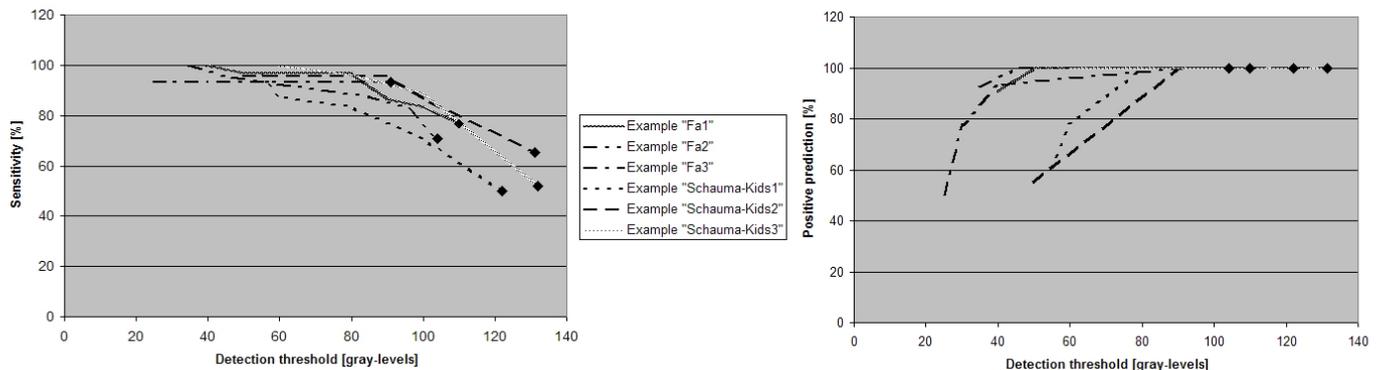


Fig. 2. Sensitivity (left) and positive predictivity (right) for images with simpler motives.

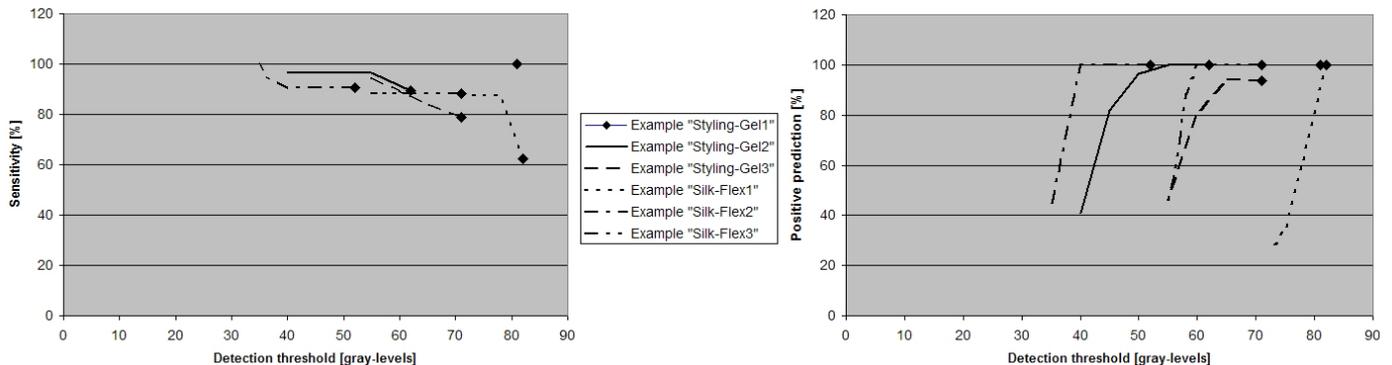


Fig. 3. Sensitivity (left) and positive predictivity (right) for images with more complex (textual) motives.

filtering. A 3×3 convolution matrix with all elements equal to 1 smooths the image, and eliminates possible unimportant discrepancies.

The array of 9 buttons at the bottom of the user interface control manual image registration, by rotating and shifting one image against the other. This option may rescue rare situations where automatic image registration could not succeed in a satisfactory image overlay.

The bottom row of buttons on the screen (bottom left in Fig. 4) turn on different presentations of the recognized printing errors. The left three correspond to the individual colour planes, while the fourth one displays all the detected errors altogether.

IV. RESULTS OF COMPUTER-AIDED ERROR DETECTION

Differences in the aligned pixel values of two registered images are caused by two sources: some of them belong to real discrepancies between images, the others, however, emerge during the process of registration (we are going to call them "phantom errors", and they are negligible from the inspection point of view). Real discrepancies can further be divided into smaller, less expressive on the one hand, and more important, actually clearly visible printing errors on the other. An inspector's attention must be drawn primarily to the latter ones.

Therefore, we decided to initially visualize only those discrepancies whose magnitude exceeds 3 standard deviations (according to the variance of image differences). Supposing the discrepancies' distribution Gaussian emphasizes about

0.3 % of the most evident errors [7]. Of course, the functionality of the user interface makes for easy lowering, or even increasing, of the threshold.

We assessed the efficiency and accuracy of the proposed printing error detection approach by experimenting, using two different methods. Firstly, we used an image and its exact copy as a pair for comparison, whereas the image copy had been corrupted by artificial, randomly added errors. The second type of experiment was completed on real image pairs whose discrepancies had been annotated by human observers. A number of image pairs was statistically assessed by sensitivity and positive predictivity, according to the threshold the image discrepancies were expected to exceed.

A. Assessment of images with artificially added errors

Two simpler and two complex images were selected, provided by our industrial partner. A pair of identical images was used for each selection, where random error regions were inserted in one of the images. The number of inserted regions varied, but no more than 30 regions were inserted, each from 1 to 10 pixels in diameter. Two kinds of errors were applied with equal probability: the first group represents a complete colour mismatch from the original colour values at randomly chosen location, while the second group encompasses errors obtained merely by random toning of the original colour values.

After adding between 60 and 70 random errors to the four given images, the sensitivity and positive predictivity



Fig. 5. Image analysis for "Fa": the errors marked with circles by a human observer (left), and automatically by our algorithm (right).



Fig. 6. Image analysis for "Silk-Flex": the errors marked with circles by a human observer (left), automatically detected by our algorithm (middle), and automatically grouped in big enough regions (right).

were calculated in three independent simulation runs, as shown in Figs. 2 and 3 left and right, respectively. The results from Fig. 2 were obtained on images with simpler, graphical contents, while the charts from Fig. 3 delineate statistics for more complex motifs containing a lot of text, such as user manuals.

Points marked with diamonds in all the charts stand for the default thresholds used in the initial visualization of the most expressive errors (upper 0.3%).

B. Assessment of real images

In contrast to the first validation attempt, pairs of images were selected here, which contained a good, referential image and an image with regions of errors that could be detected by the naked eye.

Fig. 5 shows an example of image analysis for the label "Fa". On the left, visible errors were encircled by a human observer, while on the right the detection results of our algorithm are shown as obtained using the default threshold setting of 3 standard deviations. The computer-

aided error detection discovered a lot more discrepancies than the human observer, some of them very subtle. But even more importantly, the automated detection revealed all the errors perceived by the human.

Similar results can be observed in the analysis of label "Silk-Flex", although this label appears very complex and with a lot of text on it. Fig. 6 depicts three subimages: on the left, printing errors are circled as perceived by a human observer; in the middle, the discrepancies as disclosed by our computer algorithm are marked; and on the right, the detected discrepancies are grouped and only those regions greater than a preselected area are shown.

To minimize the number of phantom and doubtfully indicated errors, two possible solutions can be followed. The first is to simply increase the threshold for the pixel differences to be detected. However, in this way some important erroneous regions might be omitted because they remain below the (too high) threshold. So, a better solution seems to be not to use higher thresholds than the default ones, but to refine detection by neglecting small discrepancy regions

TABLE I
COMPUTATIONAL COMPLEXITY [IN SECONDS]

Image [dpi]	Coarse registr.	Fine registr.	Gray-level detection	Cumulative columns 2, 3, 4	Colour detection	Gray-level + regions	Colour + regions
Fa [100]	3	9	1	14	2	1	1
Fa [200]	5	43	3	51	5	1	2
Fa [300]	11	200	8	219	11	2	4
Silk-Flex [100]	5	72	3	80	4	1	3
Silk-Flex [200]	15	136	9	160	11	2	5
Silk-Flex [300]	31	887	18	936	19	6	13

whose areas cannot be seen by the naked eye. For instance, based on [8] and with an image resolution of 300×300 dpi, such a thresholding area means 3×3 pixels. This figure was also used in the selection of the most important erroneous regions in Fig. 6, right.

C. Computational complexity of used algorithms

As one would expect, it turns out that the most time consuming part of the proposed detection algorithm is fine image registration. It comprises the affine transformation and interpolation procedures, which are applied to the individual pixels. The time needed to compute each iteration increases linearly with the number of pixels used, which, on the other hand, depends on the images' dimensions n . Therefore, the overall computational complexity is proportional to $O(n^2)$.

The times needed for image registration and error detection were measured using different image sizes. All measurements were performed on a PC with Pentium M 1600 MHz processor and 512 MB of RAM. The results are depicted in Table I for two selected images (the first with simple and the second with complex contents). Three different resolutions were taken into account (written in brackets). At first glance, quadratic computational complexity is not as obvious as one would expect, especially during the registration procedures. This is due to the different number of iterations needed to complete the registrations (it varies based on image content). But the figures fit when looking at the times needed for each iteration alone.

V. CONCLUSION

As seen in Section IV, the described approach to computer-aided error detection in printed matter proves itself to be exact and reliable. Based on the described assessments, we found that when dealing with simple images, about 95% of all real errors were detected on average, and about 90% when dealing with more complex, textual images.

Our prototype printing error detection computer program has also undergone extreme testing in an industrial environment. It was used over a couple of months during everyday inspection routine for printed matter. The most severe drawback reported was the long detection times. For example, the registration of 300 dpi images with complex textual content can take up to 15 minutes (see Table I).

A minor remark was that tiny phantom errors may appear which disturb the inspector. This problem is easily overcome by increasing the thresholds.

The problem of excessive detection duration will have to be solved by an optimization of the registration and interpolation procedures. When considering the coarse image registration, a solution in the frequency domain seems appealing, where the translational and rotational information can be extracted from the phase and amplitude spectra, respectively. Fine image registration must be optimized even more. One of the ways we are presently researching tries to minimize the quantity of data included in the computation, at each iteration. We can do this by using images with lower resolution or taking into registration only an image's segment. Afterwards, the obtained registration parameters would be applied to the original-size images. For the time being, a second optimization attempt with partial images seems the most promising.

REFERENCES

- [1] S. Periaswamy, H. Farid, Elastic Registration in the Presence of Intensity Variations, IEEE Transactions on Medical Imaging, Vol. 22, No. 7 (July 2003), pp. 865–874.
- [2] R. P. Woods, Spatial Transformation Models, In: Handbook of Medical Imaging, Isaac Bankman (Ed.), Academic Press, San Diego, 2000, pp. 465–497.
- [3] R. C. Gonzalez, Digital Image Processing: Using Matlab, 1st ed., Pearson Prentice Hall, Upper Saddle River, 2003.
- [4] QEA, IAS-1000, <http://www.qea.com/ias1000.html>
- [5] QEA, Scanner IAS, http://www.qea.com/pdf/datasheet_scannerias.pdf
- [6] FDS, FDS Imaging Software, <http://www.fdsresearch.com>
- [7] A. A. Borovkov, Mathematical Statistics, 1st ed., Gordon and Breach Science Publishers, Singapore, 1998.
- [8] B. Suhadolnik: Consultations, Henkel d. o. o., Maribor, June 2005.