Digital Ink To Form Alignment For Electronic Clipboard Devices

Jagannadan Varadarajan, Sriganesh Madhvanath

HP Laboratories
HPL-2008-131

Keyword(s):
No keywords available.

Abstract:
This paper addresses the problem of aligning handwritten input captured as digital ink to form templates, which occurs when paper forms are filled using "electronic clipboard" devices such as the ACECAD DigiMemo. Due to certain non-linearities in the digitizer hardware, the ink that is obtained from these devices is seldom in exact alignment with the "soft" form template. While not a concern for simple note capturing applications, this poses a serious problem for form processing applications. In this paper, we explore image registration approaches to solve the ink alignment problem. Two kinds of point pattern approaches to solve this alignment problem are explored. One method considers ink as "rigid" and a global transformation matrix is used to align the ink with the form template, while the second assumes only minor distortions and hence adopts a more local approach. We compare the performance of the two approaches and present some preliminary results. A summary of the work in progress and next steps are presented in conclusion.

External Posting Date: October 6, 2008 [Fulltext] Approved for External Publication
Internal Posting Date: October 6, 2008 [Fulltext]
Submitted to Document Analysis Systems (DAS), 2008 Japan.

Digital Ink To Form Alignment For Electronic Clipboard Devices

Jagannadan Varadarajan
GE Global Research
Bangalore 560066 India

Sriganesh Madhvanath
Hewlett-Packard Labs India
Bangalore 560030 India

Abstract

This paper addresses the problem of aligning handwritten input captured as digital ink to form templates, which occurs when paper forms are filled using "electronic clipboard" devices such as the ACECAD DigiMemo. Due to certain non-linearities in the digitizer hardware, the ink that is obtained from these devices is seldom in exact alignment with the "soft" form template. While not a concern for simple note capturing applications, this poses a serious problem for form processing applications. In this paper, we explore image registration approaches to solve the ink alignment problem. Two kinds of point pattern approaches to solve this alignment problem are explored. One method considers ink as "rigid" and a global transformation matrix is used to align the ink with the form template, while the second assumes only minor distortions and hence adopts a more local approach. We compare the performance of the two approaches and present some preliminary results. A summary of the work in progress and next steps are presented in conclusion.

1. Introduction

An important application of handwriting input is form filling, more so in India due to the large number of paper forms filled each year, complex scripts with inadequate keyboard support, and very low penetration of computers, internet and e-forms. We have been exploring the use of inexpensive electronic notetaking devices such as the ACECAD DigiMemo [1] (see Fig.1) for automating pen-based form filling (this device uses HP's Paperclip IP). These paper-based devices are appropriate when the end user community is not familiar with interactive devices, or when a highly distributed capture of form data needs to be carried out wherein low-cost devices are necessary. Designed primarily for offline capture of handwritten notes, these devices use standard plain paper notepads (the DigiMemo 500 series uses A5) and an inking electronic pen to deliver a hard copy as well as capture pen input (digital ink) on on-board flash storage.

Figure 1. ACECAD DigiMemo electronic clipboard

By replacing the plain paper notepad with a book of printed forms, we can create an "easy to use" form-filling application that allows the user to fill forms on paper as usual. This provides a form hardcopy, and also captures the handwritten form entries as digital ink. The digital ink captured from the forms can be uploaded to a server for processing (including recognition of some fields) and subsequent integration of the form data into an enterprise workflow, application or database. Hence, retyping of the filled forms is avoided, leading to substantial gains in efficiency and accuracy of data capture. Although devices such as the DigiMemo score high in terms of affordability, ease of use, simplicity, ruggedness and portability, a critical problem which arises when they are used for form filling is aligning the ink captured from the device with the "soft" form template. There are several hardware and environmental factors that result in the difference between (x,y) pen position recorded by the digitizer hardware and the actual pen position on the digitizing surface, Firstly, the sensor position in the pen and variations in holding the pen by users results in a pronounced offset, which the hardware neither detects nor corrects for. Secondly, small errors in printing and binding of the form booklets result in the positions of the form elements on the printed forms being slightly different from the correct positions. Lastly, the page tends to move while writing, leading to small local variations in the position of the form element on the printed page.

Digital ink processing on the server extracts ink corresponding to each form field by using the coordinates of the form element (field) in the soft form template. Due to the problems mentioned with these devices, the ink seldom falls exactly within the field. Often one or more ink strokes are "lost" outside the field, and some ink corresponding to one field ends up in another. This happens frequently with closely spaced text boxes or check boxes (See Fig.2). This can lead to incomplete or incorrect information being captured from the form.

We refer to this problem as the ink-to-form alignment problem. The problem may be avoided by the use of "self registering" paper such as Anoto Digital Pen and Paper, or expensive devices with displays such as PocketPCs and TabletPCs. However for the class of electronic ink capture devices that we are looking at, ink to form alignment is a critical problem.

In this paper we present preliminary results from using an image registration approach to solve this problem. The rest of the paper is organized as follows. In the following section we briefly review relevant literature on the related problem of form registration. In Sections 3 and 4, we present our explorations with image registration algorithms to address the ink-to-form alignment problem. In Section 5 we discuss some experimental results from these algorithms, and their shortcomings. We conclude the paper with a summary and next steps.

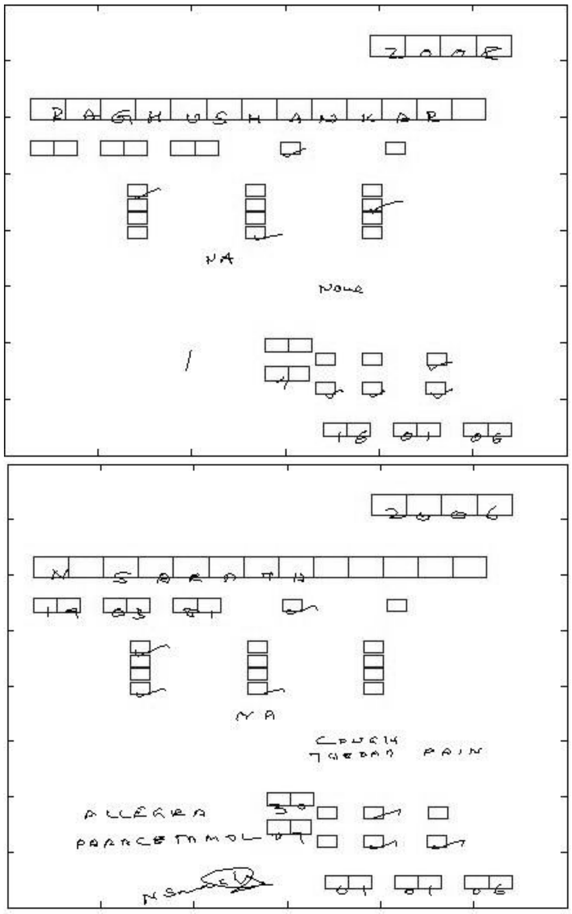

Figure 2. Examples of ink obtained from the DigiMemo overlaid on the form layout

2. Form Registration - Survey

Form registration - the task of aligning a filled form with the form template - is a key aspect of forms automation systems, and hence a well-studied problem in the area of document image processing. Form registration may be thought of as a special case of image registration - the process of matching two or more images of the same scene that are possibly taken at different times, from different sensors, or from different viewpoints. Image registration is the central sub-problem for many image processing and computer vision tasks such as image mosaicing, object tracking and so on. In general, the displacement between the contents of two images of a scene may be represented by a set of motion parameters, described using a transformation matrix called the homography matrix. Image registration works by estimating this transformation matrix by aligning different frames of a scene using a suitables matching algorithm. Form registration techniques may be broadly classified into three categories. The first is comprised of feature-based methods that use features such as lines, corner points, T junctions, fiduciary marks, logos etc, for registration purposes. The second class of methods try to learn the form type using structural features such as projection profiles, long runs of black and white pixels etc. Once the structure is understood, the bounding coordinates serve as good feature points for registration. The third class of methods that come under the featureless approach use techniques based on cross-power spectrum, log polar transforms, mutual information etc wherein, no features are extracted but the image is used to identify the registration parameters. We describe some of these methods in brief below.

Casey and Ferguson [5] presented one of the earliest works in form registration. In this method, horizontal and vertical lines were used as inputs to a neural net for form identification and further to register with their respective templates. In 1992, Taylor et al. in [18] derived a feature vector based on line crossing features from the document image, which are invariant to translation and rotation. The image is divided into nine blocks and each block is characterized by the number of the line crossings and their type. The feature vector is then used for both form identification and registration. Doerman and Rosenfield [7] in 1993 defined a basic set of constructs consisting of line segments, form regions, landmark features. Further, they classified regions as modeled and non-modeled. The modeled regions consist of those regions that contain filled in information, which is of interest. The non-modeled regions are regions of no interest. They have also presented a detailed discussion on problems resulting due to form and stroke interaction and ink detection anomalies and ink recovery. S Chandran et al. and M Garris et al. [6][8] have used structure-based techniques to identify the form type. Firstly, long runs of black pixels are identified using pixel connectivity. As most of the long runs of black pixels correspond to straight lines, the corresponding points are used as inputs to de-skewing methods such as the one proposed in [3]. Once rotation compensation is applied on the image horizontal and vertical projection profiles (histogram of black pixels in the row/column) are used to understand the structure of the form. The projection profiles also help us segregate the different fields of the form. The bounding box coordinates of the form and the various fields are then used to register with the form template. In [8], structural techniques are applied to form images down sampled to various sizes and compared. R. Safari et al. in [15][16][17] have adopted the feature based approach and in their pioneering work have proposed various affine invariants to be used for form registration. The work is based on the fundamental theorem in affine geometry with regard to uniqueness which states that there exists a unique affine transformation that maps three non-collinear points $X1, X2, X3$ into three non collinear points $Y1, Y2$, and $Y3$ respectively. Also it can be shown that an affine transformation scales the area of polygons by the same factor. Hence the ratios of the areas off two polygons considered are preserved. In [15], firstly features such as corners, fiduciary marks are extracted and for every set of four points affine invariants are calculated using all permutations of the four points. For each pair of sets of four points in both the images a minimization function of the difference of the invariants is used to arrive at the correct match. In [17], affine invariants are derived from ratios of areas of triangles, distance between line and points, and invariants involving parallel lines are used. For this Points extracted from the convex hull of the image are used and lines are extracted using Hough analysis. In [16], cross ratios involving sets of four points with respect to a fifth point is used for matching.

Under the feature less approaches, Wolberg and Zokai [22] proposed the use a log polar transform wherein image rotation and scale become simple shifts in x and y directions. Kuglin and Hines [13] used phase correlation for translational registration. Hutchison et al. in [11] addressed the problem adopting the Fourier-Mellin Transform. This method determines the translation, rotation, scale and shear parameters that map one image to another. The proposed method overcomes the inherent disadvantages of Fourier-Mellin transform by exploiting the properties of forms, which are tabular in nature. Tabular documents are replete with horizontal and vertical lines. These lines are similar to periodic wave fronts and give rise to dominant peaks in the frequency domain that are perpendicular to each other. The most periodic linear components will produce the strongest line over the power spectrum. This makes it easy to find the angle of rotation. It is possible to decouple the two axes and identify the skew angle as well as the shear angle.

In order to solve the ink to form alignment problem, we looked into some of these approaches from the literature on form registration. As opposed to the normal form registration problem wherein two form images are the inputs, the ink-form alignment problem has as inputs (i) the digital ink corresponding to the form input on the one hand, and (ii) the "soft" form template on the other. The digital ink for the purposes of this paper is assumed to be represented as a sequence of (x, y) points corresponding to individual ink strokes (defined by consecutive pen-down and pen-up events) in a simple ASCII representation such as UNIPEN [2], and the form template as a sequence of $(x, y, width, height)$ tuples in a simple CSV format.

In the next two sections, we describe some of the approaches we have explored. These are based on the image registration literature, where as described earlier, the central problem is to compute a $homography$ matrix that describes the transformation between the two images.

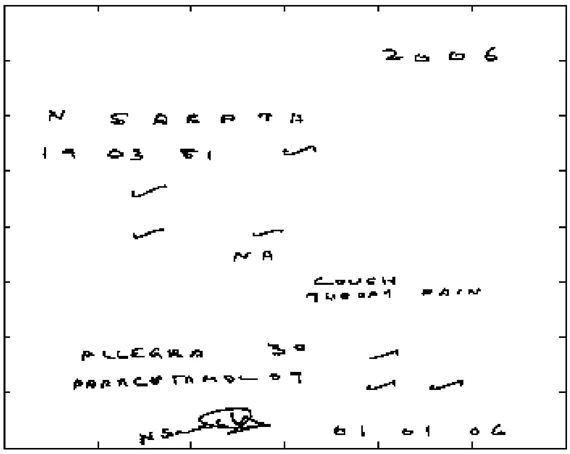

Figure 3. Ink strokes from a form shown above

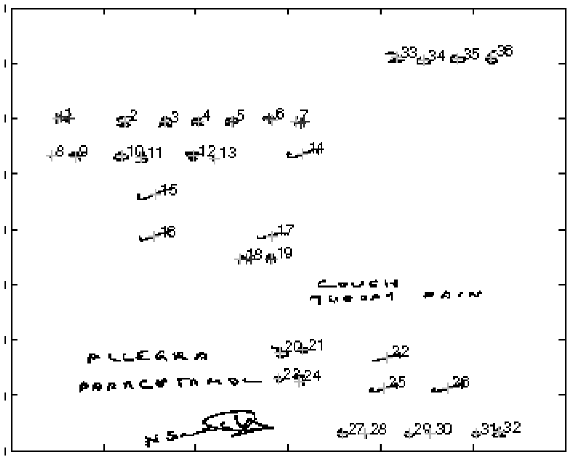

Figure 4. Representatives selected from each group of strokes. Only numbered/marked strokes are considered for computation of alignment. Unmarked ink strokes correspond to drawing areas and are ignored

3. Global Point Pattern Approach

In our scenario, since we do not have images with any common or even similar information, it is not possible to detect feature points using algorithms such as KLT [20] Harris [9]. But it is certainly possible to derive a point set from the ink data, to align with a point set derived from the form layout file. We accomplish this by considering the centers of the bounding boxes of ink strokes (See Fig.3 for ink strokes obtained from the form), and the centers of boxes on the form, as the point sets from the ink file and the form layout respectively. In the case of multi-stroke characters, considering all the stroke centers as input for the registration algorithm poses a problem since the registration algorithm only offers a one to one mapping of point sets. We address this problem by grouping nearby strokes into hypothetical characters based on a proximity criterion. For each group, one stroke is selected as its representative. In addition, strokes corresponding approximately to drawing areas and free text form fields (e.g. free-form writing and signatures) are ignored for the purposes of computing the transformation.See Fig.4, Only strokes that are numbered are considered for computation of the alignment. Strokes that are not numbered/marked correspond to drawing areas like signature etc and are hence ignored

In our scenario, we expect to see distortions of translation and rotation, but not higher degree transformations such as scale, shear and perspective. This reduces the problem to matching the point sets and estimating the planar transformation present. Since the computation of descriptors from feature points as in [12] is not possible, we initially adopted the point pattern approach [21] for matching the ink and the form fields.

The basic algorithm that we chose to implement was taken from the work by Wamelan et al. in [21]. The Point Pattern approach is used when the misalignment is due to large rotation and translation. Consider two points sets P and Q, where Q contains a subset of points from P transformed by a affine transformation matrix T. The idea of the algorithm is that with high probability, one of the first few random points in P after the transformation T will correspond to some points in Q. We try to look for an affine transformation that maps the nearest neighbors of a point in P to those of Q, locally. If we find such a map, it is easy to check whether it also gives a "global" match. For our scenario, the algorithm proposed by [21] was implemented as a two-point pattern algorithm, where we look for patterns formed by pairs of points. By mapping the pairs of points, an estimate of the transformation parameters is obtained. After deriving the local estimate of the transformation, a global estimate of the parameters is arrived at by applying the local transformation to all the points in P and scoring the number of points the transformation maps from P to Q. The $homography$ matrix obtained thus is applied to all the ink strokes. The results obtained by applying the point pattern registration algorithm to the ink strokes are shown in Fig.5. The figure shows the transformed ink strokes overlaid on the form layout to enable visual inspection of the algorithm's performance.

4. Local Point Pattern Approach

From the examples in Fig.5, it can be seen that while most of the ink strokes are aligned within the box, a few still remain unaligned. This may be attributed to the fact that the algorithm provides a single estimate of the transformation parameters. While this global transformation succeeds in aligning most of the ink strokes with the corresponding boxes, it does not sufficiently compensate for all local variations resulting from various factors described in the introductory section. For instance, while a few of the ink strokes have an offset in the positive x, y direction, others have offsets in the negative x, y direction making a single, global transformation insufficient. Moreover, this also requires a post-processing step to align all the ink into the boxes perfectly. This could be done by sorting the points in both the point sets with respect to their coordinates and then perform a merge like pass between the two sets.

We therefore decided to explore more local approaches. We explored a new algorithm, which works point-wise and estimates the transformation more locally. This algorithm considers each pair of points, along with the Euclidean distance between them as a (very simple) pattern, and constructs sets of all possible patterns from both the point sets.

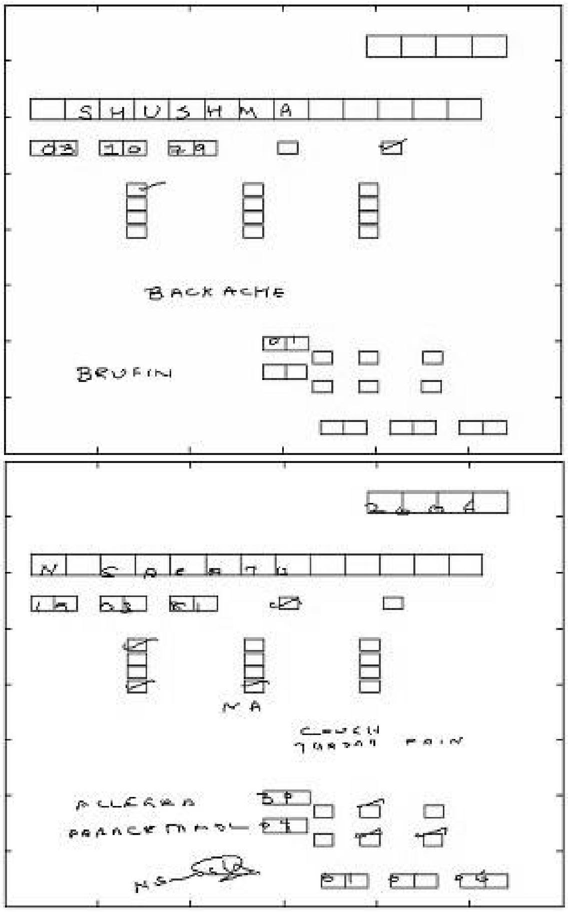

Figure 5. Ink after registration using the global point pattern matching approach

For a particular pattern (represented by a pair of points) all possible matches in the other pattern set is looked at. In order to reduce the time complexity of the search we take advantage of the assumption that the distance between the ink stroke center and the box field center denoted by "d" can be estimated as a function of the box dimensions $d = f(l, b)$, where l and b are length and breadth respectively of the smallest box in the form layout. Therefore the search space of possible matches for a box field is constrained by searching within a radius d from the box center. An exhaustive set of possible mappings is arrived at after this constrained search. From this set of putative matches, a consistent one-to-one mapping is obtained by constructing an accumulator matrix M and scoring the matches. An ink stroke is said to match a box center only if the corresponding accumulator value is highest both row wise and column wise. Once an ink stroke is assigned to a box, the entire group of strokes represented by that stroke is centered into the box.

4.1. Algorithm

Let P_{ij} be set of all patterns formed by pairs of points p_i and p_j in the point set P. Let Q_{mn} be set of all patterns formed by pairs of points q_m and q_n in the point set Q. Let M be a matrix of size $|P| \times |Q|$ where $|P|$ and $|Q|$ are sizes of set P and Q. we call it the accumulator matrix which contains the scores of the mapping.

For every pair in P_{ij}
    For every pair in Q_{mn}
        If the ratio of distances $d(p_i, p_j)$ and $d(q_m, q_n)$ is close to unity
            If $(|q_m - p_i| < d$ and $|q_n - p_j| < d)$
                then map q_m to p_i and q_n to p_j
                increment the entries (i, m) and (j, n) in the matrix M
            elseif $(|q_m - p_j| < d$ and $|q_n - p_i| < d)$
                then map q_m to p_j and q_n to p_i
                Increment the entries (i, m) and (j, n) in the matrix M

Derive the mapping from the accumulator matrix

- A zero value in the accumulator matrix indicates that the particular ink stroke has no match and therefore can be discarded.
- If one stroke becomes a candidate for two boxes, Calculate the distance between the box and stroke and assign the stroke to the box with shortest distance
- If more than one stroke are possible candidates for a box
  - If the scores in the matrix M are equal, assign the stroke to the box with the shortest distance to the stroke.
  - If the scores are unequal consider the stroke with maximum score and zero out the other stroke
- Check to ensure a one-one mapping
- For every $(inkstroke, boxcenter)$ pair obtained from the mapping, calculate the offset in the form of x, y values and apply the difference on the ink coordinates

5. Experimental Results and Discussion

The two alignment algorithms were tested on a dataset of 60 ink files corresponding to three different forms, and collected using three different DigiMemo devices from various colleagues, and the results were inspected by rendering the corrected ink strokes on the form template. Strokes whose centroids fall outside the form element/box are considered unaligned. Fig.7 shows some results from applying the local point-pattern algorithm described in section 4 to the ink samples. From these examples, it is clear that the local approach performs better in aligning more number of ink strokes with the corresponding form boxes. Fig.6 provides the number of characters entered by the user for a particular form type, and the number of characters remaining unaligned after global point-pattern approach, and local point pattern approach.

Form No	Number of Characters entered in the Form	Number of unaligned characters after Global & Local Point Pattern Approach	
		Global	Local
1	69	2	5
2	69	0	0
3	73	0	0
4	73	5	0
5	73	16	0
6	73	36	0
7	80	2	0
8	79	24	0
9	79	30	1
10	46	0	0
11	36	2	0
12	68	13	1
13	67	0	0
14	67	4	0
15	74	0	1

Figure 6. Number of errors (unaligned characters) after global Vs local point pattern algorithm for a particular form type

These preliminary results suggest that the local approach is superior compared to its global counterpart. This is largely due to the fact that the distortions we have in our dataset are (i) fairly local, and (ii) different for different fields. There are a number of other types of distortions as described in the introduction that are not captured in our dataset which may require other methods. In particular, it is likely that replacing the bound form booklet with loose leaf forms may require the use of the global and local alignment approaches in sequence.

One obvious shortcoming of the local approach is that the alignment it computes for an ink stroke may be too local. For example, since an ink stroke is essentially centered into the corresponding box, the distinction between an apostrophe and a comma when written separately in a box is essentially lost. We believe that computing alignment at the level of a form field may provide a balance between global and local extremes. Second areas of possible improvement are the proximity criterion used to group strokes into hypothetical characters and select a representative stroke from the group. The present greedy approach may result in grouping errors especially when boxes are bunched close together on the layout. For local matching, an alternative to the local point pattern approach may be bipartite graph matching, which has found wide application in wide base line correspondence [19] and shape matching [4]. In our case, it is possible to construct a bipartite graph with the features from the ink and the form template as the two sets of vertices. The matching could be obtained using methods like the Hungarian algorithm [14] or the Hopcroft and Karp [10].

6. Summary and Next Steps

In this paper we have presented the problem of ink-form alignment problem that is encountered while using Electronic clipboard devices for form filling applications. The various causes of the problem and its effects on form recognition are discussed. Two approaches inspired by image registration - Global Point Pattern and Local Point Pattern - were proposed to address the problem. A comparison of the two methods suggests that the local approach is superior compared to its global counterpart, at least on the dataset used for our experiments. There is considerable room for further improvements to these methods, as discussed in the previous section. Further we also intend to explore and compare in the future bipartite graph matching methods with the local point pattern method for aligning ink with form fields.

References

[1] Product information on acecad digimemo 502. http://www.acecad.com.tw/dma502.html.

[2] Unipen 1.0 format definition. www.unipen.org.

[3] H. Baird. The skew angle of printed documents. *Document Image Analysis*, pages 204–208, April 1995.

[4] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *Pattern Analysis and Machine Intelligence*, 24(4), April 2002.

[5] R. Casey, D. Ferguson, K. Mohiuddin, and E. Walach. Intellingent forms processing. *Machine Vision Applications*, 5(3):143–155, June 1992.

[6] S. Chandran and R. Kasturi. Structural recognition of tabulated data. In *Proc. of the Intl Conf. on Document Analysis and Recognition*, volume 5, pages 516–519, October 1993.

[7] D. S. Doermann and A. Rosenfeld. The processing of form documents. In *Proc. of the Intl Conf. on Document Analysis and Recognition*, volume 5, pages 497–501, October 1993.

[8] M. Garris and P. Grother. Structural recognition of tabulated data. *Document Analysis and Information retrieval*, pages 321–334, April 1996.

[9] C. Harris and M. Stephens. A combined corner and edge detector. In *Proc Alveyvision conference*, pages 147–151, August 1988.

[10] J. Hopcroft and R. Karp. An n5/2 algorithm for maximum matchings in bipartite graphs. *SIAM Journal on Computing*, 2, August 1972.

[11] L. A. D. Hutchison and W. A. Barrett. Fast registration of tabular document images using the fourier-mellin transform. In *Proc of the First Intl Workshop on Document Image Analysis for Libraries*, pages 253–267, August 2004.

[12] V. Jagannadan, M. C. Prakash, R. R. Sarma, and G. P. Rao. Feature extraction and image registration of color images using fourier bases. In *IEEE Intl Conf on Image Processing*, volume 2, pages 662–665, October 2005.

[13] C. Kuglin and D. Hines. The phase correlation image alignment method. In *IEEE Conference on Cybernetics and Society*, volume 2, pages 163–165, August 1975.

[14] H. Kuhn. A hungarian method for the assignement problem. *Naval Research Logistics Quaterly*, August 1955.

[15] R.Safari, N.Narasimhamurthi, M. Shridhar, and M. Ahmadi. Affine point pattern matching and its applications to form registration. In *IEEE Intl Symposium on Circuits and Systems*, volume 2, pages 656–659, May 1996.

[16] R.Safari, N.Narasimhamurthi, M. Shridhar, and M. Ahmadi. Document registration using projective geometry. In *Intl. Conf. on Document Analysis and Recognition*, volume 6, pages 1337–1341, September 1997.

[17] R.Safari, N.Narasimhamurthi, M. Shridhar, and M. Ahmadi. Form registration: a computer vision approach. In *Proc. of the Fourth Intl Conf on Document Analysis and Recognition*, volume 2, pages 758–761, August 1997.

[18] S. L. Taylor, R. Fritzson, and J. A. Pastor. Extraction of data from preprinted forms. *Machine Vision and Applications*, 5(3):211–222, June 1992.

[19] D. Tell. *Widebaseline spectral matching with applications to visual serving*. PhD thesis, University of Stockholm, 2002.

[20] C. Thomasi and T. Kanade. Detection and tracking of point features. *CMU Technical Report*, pages 593–600, April 1991.

[21] P. B. V. Wamelen, Z. LI, and S. S. Iyengar. A fast expected time algorithm for the 2-d point pattern matching problem. *Pattern Recognition*, 37(8):1699–1711, 2004.

[22] G. Wolberg and S. Zokai. Robust image registration using log-polar transform. In *Proc. IEEE Intl. Conf. on Image Processing*, September 2000.

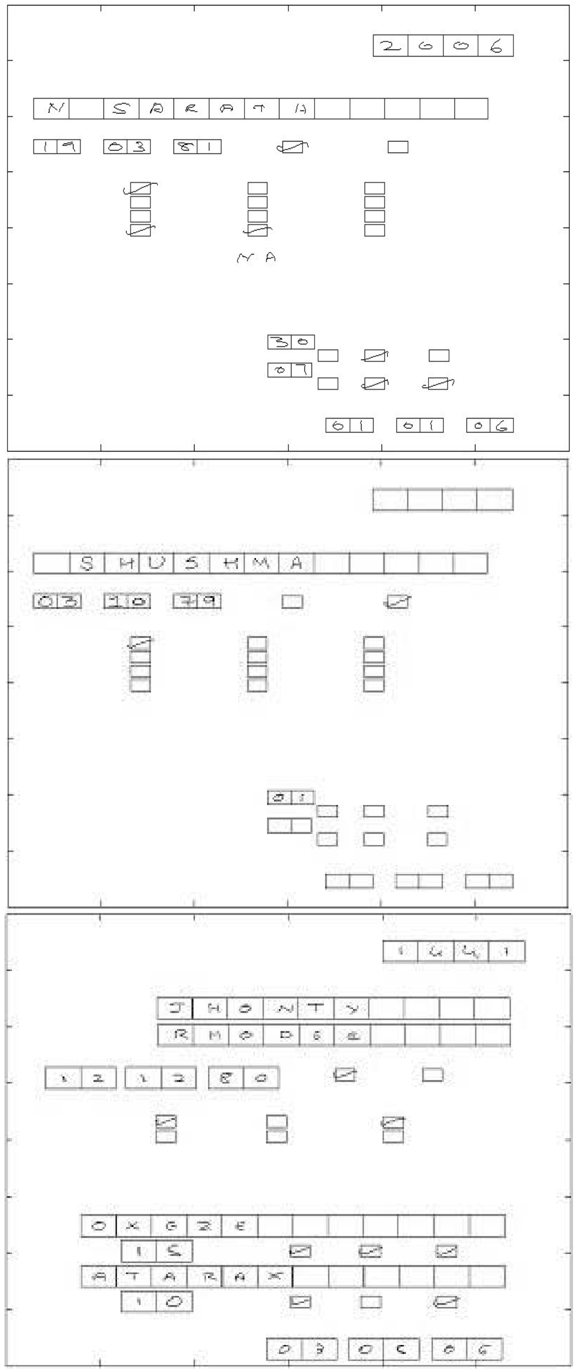

Figure 7. Ink after registration using the Local point pattern matching approach applied to two different form types