LAMP-TR-
CAR-TR-
CS-TR-

October 1998

An Automatic Closed-Loop Methodology for Generating Character Groundtruth

Tapas Kanungo
Robert M. Haralick

An Automatic Closed-Loop Methodology for Generating Character Groundtruth

Tapas Kanungo
Robert M. Haralick

Center for Automation Research
University of Maryland
College Park, MD 20742
kanungo@cfar.umd.edu

Department of Electrical Engineering
University of Washington
Seattle, WA 98195, USA
haralick@george.ee.washington.edu

Abstract

Character groundtruth for real, scanned document images is extremely useful for evaluating the performance of OCR systems, training OCR algorithms, and validating document degradation models. Unfortunately, manual collection of accurate groundtruth for characters in a real (scanned) document image is not possible because (i) accuracy in delineating groundtruth character bounding boxes is not high enough, (ii) it is extremely laborious and time consuming and (iii) the manual labor required for this task is prohibitively expensive.

In this paper we give a closed-loop methodology for collecting very accurate (within a pixel error) groundtruth for scanned documents. We first create ideal documents using a typesetting language. Next we create the groundtruth for the ideal document. The ideal document is then printed, photocopied and then scanned. A registration algorithm estimates the global geometric transformation and then performs a robust local bitmap match to register the ideal document image to the scanned document image. Finally, groundtruth associated with the ideal document image is transformed using the estimated geometric transform to create the groundtruth for the scanned document image.

This methodology is very general and can be used for creating groundtruth for documents in typeset in any language, layout, font, and style. We have demonstrated the method by generating groundtruth for scanned and FAXed images. The cost of creating groundtruth using our methodology is minimal. If character, word or zone groundtruth is available for any real document, the registration algorithm can be used to generate the corresponding groundtruth for a rescanned version of the document.

Keywords: Automatic real groundtruth, document analysis, performance evaluation, registration, geometric transformations, image warping.

1 Introduction

Character groundtruth for real, scanned document images is extremely useful for evaluating the performance of OCR systems, training OCR algorithms, and validating document degradation models. Unfortunately, manual collection of accurate groundtruth for characters in a real (scanned) document image is not possible because (i) accuracy in delineating groundtruth character bounding boxes is not high enough, (ii) it is extremely laborious and time consuming and (iii) the manual labor required for this task is prohibitively expensive. In fact, successful OCR companies have large number of employees whose only job is to manually annotate scanned documents with the ground truth.

In this paper we present a closed-loop methodology for collecting very accurate groundtruth for scanned documents. We first create ideal documents using a typesetting language. Next we create the groundtruth for the ideal document. The ideal document is then printed, photocopied and then scanned. A registration algorithm estimates the geometric transformation that registers the ideal document image to the scanned document image. Finally, groundtruth associated with the ideal document image is transformed using the estimated geometric transform to create the groundtruth for the scanned document image.

The groundtruth generated by this method, besides being directly useful for evaluating the performance of OCR systems, is crucial for validating document degradation models [KHB+94, KBH95]. In fact, manually collected groundtruth invariably contains many outliers and forces the use of robust statistical techniques [KHB95]. For a more detailed discussion please see [Kan96]. Parts of this work was earlier reported in [KH96].

We are unaware of any literature that uses a method similar to ours for automatically collecting groundtruth. Certainly lot of work on document registration has been reported in the past. However, most of this literature pertains to the problem where a fixed ideal form has to be registered to a scanned, hand-filled form. The general idea is to extract the information filled by a human in the various fields of the form. A common method is to extract features from the scanned forms and match them to the features in the ideal form [DR93, CF90]. Unfortunately we cannot use this body of work since there are no universal landmarks that appear at fixed locations in each type of document.

2 Document Groundtruth

Groundtruth information is essential for evaluating any document understanding systems. By *groundtruth* we mean the correct location, size, font type, and bounding box of the individual symbols on the document image. More global groundtruth associated with a document image could include layout information (such as zone bounding boxes demarcating individual words, paragraphs, article and section titles, addresses, footnotes etc.) and style information (general information regarding number of columns, right justified or not, running head; etc). The groundtruth information, of course, needs to be 100 percent accurate, otherwise the systems being evaluated will be penalized incorrectly.

Groundtruth information is invaluable for performance evaluation of OCR algorithms. Use of completely annotated groundtruth permits us to study which factors affect the algorithm performance the most. This in turn allows an algorithm developer to think of ways to improve the algorithm. Among the dimensions of complete annotation are layout, style, font type, font size, symbol location, and symbol identity.

Style: Page numbers can be printed on top or bottom; the document may or may not have a runninghead; various indentation lengths can be varied; the columns can be justified or ragged; number of columns can be changed. Thus by changing these variables, we can study how robust the OCR system is with respect to these style parameters.

Font: OCR systems can be very sensitive to the fonts used. Thus we can study the performance of the OCR algorithm by changing the various fonts (Helvetica, Times Roman, etc.) used in the documents, while keeping the text unchanged. Furthermore, OCR algorithms usually have a subsystem that identifies the font in a particular zone. Performance of such systems can be done if the groundtruth information about the fonts is available.

Size: Just as some OCR systems have subsystems that identify the font types, other OCR systems have subsystems that identify character size, which then is used by the recognition engine. Having the bounding box information associated with each symbol on the page will allow us to evaluate the performance of these subsystems.

Location and Identity: Finally, since the groundtruth contains the location and identity (e.g., which character, or math symbol) of each symbol on the page, we can use this information to evaluate the performance of the OCR system.

In our previous work [KHP93, KHP94, HP+94], we showed how very accurate groundtruth for ideal documents images can generated. We also showed that the groundtruth for ideal document images could also be used as groundtruth for synthetically degraded documents. The actual process was as follows. We first created noise-free document images using the LaTeX typesetting language [Knu88, Lam86], and extracted the groundtruth information from the DVI files generated by LaTeX. Then we then synthetically degraded the ideal binary document image using a local document degradation model. The the groundtruth for the synthetically degraded documents is 100% accurate, is easily generated (few seconds on SPARC 5), and does not cost anything once we have the LaTeXfiles.

Unfortunately, once the ideal document image is printed and scanned, the groundtruth information associated with the ideal document image is not usable for the scanned document image since the scanned document is geometrically transformed. That is, the printing and scanning process, to a first order effect, translates, scales and rotates each character on the document image, besides adding pixel noise. The second order effect on each character is translated by a few pixels from its nominal position after the translation, rotation and scale.

Thus the only alternative is to manually enter the groundtruth for the real, scanned documents. This task s extremely laborious, time consuming and prohibitively expensive. Furthermore, the person creating the groundtruth should be knowledgeable in the language in which the document is written.

In the following section we outline an automatic, closed-loop approach to generation of groundtruth for real documents. This methodology is very general and is independent of the language in which the document text is written.

3 The groundtruth generation methodology: a closed-loop approach

First, the documents are typeset using LaTeX. Next these documents are converted into binary bitmap images, which are our ideal noise-free documents. When the ideal bitmap is generated from the DVI files, the corresponding groundtruth (location, bounding box, font type and size, and identity of each character) is also generated.

The ideal document image is then printed and scanned. At this point, although the groundtruth for the ideal image is known, the groundtruth for the real scanned image is unknown. However, if the exact transformation that registers the ideal and degraded images were known, we could compute the groundtruth for the real image simply by transforming the bounding box coordinates of the ideal groundtruth by the same transformation.

Thus the groundtruth creation problem now reduces to finding an appropriate transformation that models the geometric distortions the document image undergoes when it is printed and then scanned. Whatever the functional form of the transformation, to estimate the parameters of the transformation we require corresponding feature points from the ideal and real images. Thus, a rough outline of the groundtruth generation method is:

1. Generate ideal document images and the associated character groundtruth.
2. Print the ideal documents and scan it back.
3. Find feature points $p_1, \ldots, p_n$ and $q_1, \ldots, q_n$ in the corresponding ideal and real document images.
4. Establish the correspondence between the points p_i and q_i.
5. Estimate the parameters of the transformation T that maps p_i to q_i.
6. Transform the ideal groundtruth information using the estimated transformation T.

The transformation T mentioned in the procedure above is a $2D$ to $2D$ mapping. That is $T : R^2 \rightarrow R^2$. Thus, if $(x, y) = T(u, v)$, where (u, v) is the ideal point and (x, y) is the scanned point, x in general may be a function of both u and v; and same is true regarding y.

Generation of the ideal document image and the corresponding groundtruth is achieved by the synthetic groundtruth generation software DVI2TIFF, which was described in [Kan96] The software is available with the UW English Document Database [HP$^+$94]. Given a transformation T, transforming the groundtruth information is trivial – all that needs to be done is transform the bounding box coordinates of the ideal groundtruth using the transformation T. Thus, there are two main problems: finding corresponding feature points in two document images, and finding the transformation T.

4 Geometric Transformations

Suppose we are given the coordinates of feature points p_i on an ideal document page, and the coordinates of the corresponding feature points q_i on the real document page. (How these feature points are extracted and matched is described in section 6.) The problem is to hypothesize a functional form for the transformation T, that maps the ideal feature points coordinates to the real point coordinates, and a corresponding noise model. To ensure that the transformation T is the same throughout the area of the document page, we choose the points p_i from all over the document page.

The possible candidates for the geometric transformation and pixel perturbation are similarity, affine, and projective transformations:

Similarity Transformation: This transformation is defined by the equation:

$$\begin{pmatrix} x_i \\ y_i \end{pmatrix} = \begin{pmatrix} a & b \\ -b & a \end{pmatrix} \cdot \begin{pmatrix} u_i \\ v_i \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} + \begin{pmatrix} \eta_i \\ \psi_i \end{pmatrix} \tag{1}$$

where (u_i, v_i) is the ideal point, (x_i, y_i) is the transformed point, and $(\eta_i, \psi_i)^t \sim N(0, \sigma^2 I)$ is the noise.

In the above parameterization of the similarity transformation, a represents the product of a nonnegative isotropic scale and cosine of the rotation angle; b represents the product of the nonnegative scale and sine of the rotation angle; t_x and t_y represent the translation in x and y directions. This parametrization is linear and unconstrained in the parameters, unlike the parametrization in terms of scale, cosine and sine of rotation angle, and translation.

Affine Transformation: In this case we assume that the real image is an affine transformation of the ideal image. The affine transformation allows for rotation, translation, anisotropic scale, and shear. The functional form that maps the ideal point onto the real point is:

$$\begin{pmatrix} x_i \\ y_i \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot \begin{pmatrix} u_i \\ v_i \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} + \begin{pmatrix} \eta_i \\ \psi_i \end{pmatrix} \tag{2}$$

where (a, b, c, d, e, f) is the affine transform parameters, (u_i, v_i) is the ideal point, (x_i, y_i) is the transformed point, and $(\eta_i, \psi_i)^t \sim N(0, \sigma^2 I)$ is the noise.

Projective Transformation: Here the assumption is that the real image is a perspective projection of an image on a plane onto another nonparallel plane. The functional form that maps the ideal point (u_i, v_i) onto the real point (x_i, y_i) is

$$\begin{pmatrix} x_i \\ y_i \end{pmatrix} = \frac{1}{a_3 u_i + b_3 v_i + 1} \begin{pmatrix} a_1 u_i + b_1 v_i + c_1 \\ a_2 u_i + b_2 v_i + c_2 \end{pmatrix} + \begin{pmatrix} \eta_i \\ \psi_i \end{pmatrix} \tag{3}$$

where, (u_i, v_i) is the ideal point, (x_i, y_i) is the transformed point, and $(\eta_i, \psi_i)^t \sim N(0, \sigma^2 I)$ is the noise. After inspection it can be seen that the equations are linear and unconstrained in the eight parameters $a_1, b_1, c_1, a_2, b_2, c_2, a_3, b_3$. Discussion on the estimation of these parameters can be found in subsection 5. This parameterization accounts for rotation, translation and the center of perspectivity parameters.

In the above discussion σ can be assumed to be known and a function of the spatial quantization error and the image processing algorithm that is used to detect the feature points.

Each of these models can be used to fit the data. Nevertheless, the question is which model, if any, models the transformations correctly, and how does one go about proving the hypothesis quantitatively?

In the next section, we show how to estimate the parameters of the three models. In the section that follows we show how to statistically validate/invalidate the models.

5 Estimation of geometric transformation parameters

Note that all the three models are linear in the parameters. Each corresponding point provides two linear constraints on the parameters. Thus we need at least two corresponding points for estimating the similarity parameters, three corresponding points for affine, and four for projective. If we have more corresponding points than the minimum required, we can solve for the parameters of the transformation in a least squares sense, which also happens to be the maximum likelihood estimate of the parameters under the Gaussian noise model.

5.1 Similarity transformation

The similarity equations given in equation (1) can be rearranged into the following form:

$$\begin{pmatrix} x_i \\ y_i \end{pmatrix} = \begin{pmatrix} 1 & 0 & u_i & v_i \\ 0 & 1 & v_i & -u_i \end{pmatrix} \cdot \begin{pmatrix} t_x \\ t_y \\ a \\ b \end{pmatrix} + \begin{pmatrix} \eta_i \\ \psi_i \end{pmatrix}, \tag{4}$$

where (u_i, v_i) is the ideal point, (x_i, y_i) is the transformed point, (a, b, t_x, t_y) are the similarity transform parameters, and (η_i, ψ_i) is the noise. If there are n corresponding points, the above equation can be written as:

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \\ y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & 0 & u_1 & v_1 \\ 1 & 0 & u_2 & v_2 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & u_n & v_n \\ 0 & 1 & v_1 & -u_1 \\ 0 & 1 & v_2 & -u_2 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 1 & v_n & -u_n \end{bmatrix} \cdot \begin{pmatrix} t_x \\ t_y \\ a \\ b \end{pmatrix} + \begin{bmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_n \\ \psi_1 \\ \psi_2 \\ \vdots \\ \psi_n \end{bmatrix}. \tag{5}$$

The above equation can be written in a compact matrix form as follows:

$$\mathbf{b} = \mathbf{A}\mathbf{p} + \mathbf{n} \tag{6}$$

where $\mathbf{b}$ is the $2n \times 1$ vector of x and y, $\mathbf{A}$ is the $2n \times 4$ form matrix, $\mathbf{p}$ is the 4×1 vector of unknown parameters, and $\mathbf{n}$ is the $2n \times 1$ vector of noise values. If the number of corresponding points n is two, we have four equations in four unknowns, and thus can solve for $\mathbf{p}$ uniquely by solving the system of equations:

$$\mathbf{b} = \mathbf{A}\hat{\mathbf{p}}. \tag{7}$$

However, if we have more than two correspondences, we can solve for $\mathbf{p}$ in a least squares sense by solving the following linear system of equations [Kan96]:

$$\mathbf{A}^t\mathbf{b} = \mathbf{A}^t\mathbf{A}\hat{\mathbf{p}}. \tag{8}$$

5.2 Affine transformation

The affine equations given in equation (2) can be rearranged into the following form:

$$\begin{pmatrix} x_i \\ y_i \end{pmatrix} = \begin{pmatrix} u_i & v_i & 0 & 0 & 1 & 0 \\ 0 & 0 & u_i & v_i & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} a \\ b \\ c \\ d \\ e \\ f \end{pmatrix} + \begin{pmatrix} \eta_i \\ \psi_i \end{pmatrix}. \tag{9}$$

where (u_i, v_i) is the ideal point, and (x_i, y_i) is the transformed point, (a, b, c, d, e, f) are the affine transform parameters, and (η_i, ψ_i) is the noise.. If there are n corresponding points, the above equation can be written as:

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \\ y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} u_1 & v_1 & 0 & 0 & 1 & 0 \\ u_2 & v_2 & 0 & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_n & v_n & 0 & 0 & 1 & 0 \\ 0 & 0 & u_1 & v_1 & 0 & 1 \\ 0 & 0 & u_2 & v_2 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & u_n & v_n & 0 & 1 \end{bmatrix} \cdot \begin{pmatrix} a \\ b \\ c \\ d \\ e \\ f \end{pmatrix} + \begin{bmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_n \\ \psi_1 \\ \psi_2 \\ \vdots \\ \psi_n \end{bmatrix}. \tag{10}$$

The above equation can be written in a compact matrix form as follows:

$$\mathbf{b} = \mathbf{A}\mathbf{p} + \mathbf{n} \tag{11}$$

where $\mathbf{b}$ is the $2n \times 1$ vector of x and y, $\mathbf{A}$ is the $2n \times 6$ form matrix, $\mathbf{p}$ is the 6×1 vector of unknown parameters, and $\mathbf{n}$ is the $2n \times 1$ vector of unknown noise values. If the number of corresponding points n is three, we have six equations in six unknowns, and thus can solve for $\mathbf{p}$ uniquely by solving the system of equations

$$\mathbf{b} = \mathbf{A}\hat{\mathbf{p}}. \tag{12}$$

If we have more than three correspondences, we can solve for $\mathbf{p}$ in a least squares sense by solving the following linear system of equations [Kan96]:

$$\mathbf{A}^t\mathbf{b} = \mathbf{A}^t\mathbf{A}\hat{\mathbf{p}}. \tag{13}$$

5.3 Projective transformation

The projective transformation equations given in equation (3) can be rearranged in the following form.

$$\begin{pmatrix} x_i \\ y_i \end{pmatrix} = \begin{pmatrix} u_i & v_i & 1 & 0 & 0 & 0 & -u_i x_i & -v_i x_i \\ 0 & 0 & 0 & u_i & v_i & 1 & -u_i y_i & -v_i y_i \end{pmatrix} \cdot \begin{pmatrix} a_1 \\ b_1 \\ c_1 \\ a_2 \\ b_2 \\ c_2 \\ a_3 \\ b_3 \end{pmatrix} + \begin{pmatrix} \eta_i \\ \psi_i \end{pmatrix}, \tag{14}$$

where (u_i, v_i) is the ideal point, (x_i, y_i) is the transformed point, $(a_1, b_1, c_1, a_2, b_2, c_2, a_3, b_3)$ are the projective transform parameters, and (η_i, ψ_i) is the noise.. transformed point. If there are n corresponding points, the above equation can be written as:

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \\ y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} u_1 & v_1 & 1 & 0 & 0 & 0 & -u_1 x_1 & -v_1 x_1 \\ u_2 & v_2 & 1 & 0 & 0 & 0 & -u_2 x_2 & -v_2 x_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_n & v_n & 1 & 0 & 0 & 0 & -u_n x_n & -v_n x_n \\ 0 & 0 & 0 & u_1 & v_1 & 1 & -u_1 y_1 & -v_1 y_1 \\ 0 & 0 & 0 & u_2 & v_2 & 1 & -u_2 y_2 & -v_2 y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & u_n & v_n & 1 & -u_n y_n & -v_n y_n \end{bmatrix} \cdot \begin{pmatrix} a_1 \\ b_1 \\ c_1 \\ a_2 \\ b_2 \\ c_2 \\ a_3 \\ b_3 \end{pmatrix} + \begin{bmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_n \\ \psi_1 \\ \psi_2 \\ \vdots \\ \psi_n \end{bmatrix} \tag{15}$$

The above equation can be written in a compact matrix form as follows.

$$\mathbf{b} = \mathbf{A}\mathbf{p} + \mathbf{n} \tag{16}$$

where $\mathbf{b}$ is the $2n \times 1$ vector of x and y, $\mathbf{A}$ is the $2n \times 8$ form matrix, $\mathbf{p}$ is the 8×1 vector of unknown parameters, and $\mathbf{n}$ is the $2n \times 1$ vector of noise values. If the number of corresponding points n is four, we have eight equations in eight unknowns, and thus can solve for $\mathbf{p}$ uniquely by solving the following system of equations:

$$\mathbf{b} = \mathbf{A}\hat{\mathbf{p}}. \tag{17}$$

If we have more than four correspondences, we can solve for for $\mathbf{p}$ in a least squares sense by solving the following linear system of equations (see [Kan96]):

$$\mathbf{A}^t\mathbf{b} = \mathbf{A}^t\mathbf{A}\hat{\mathbf{p}}. \tag{18}$$

6 Finding corresponding feature points

In a document image with text, figures and mathematics, there are no universal feature points in the interior of the document that are guaranteed to appear in each type of document. However, most documents have a rectangular text layout, whether they are in one-column format or in two-column format. We use the upper-left (UL), upper-right (UR), lower-right (LR), and lower-left (LL), corners of the text area as feature points.

The four feature points, $p_1, \ldots, p_4$, are detected on the ideal image as follows (assume the origin at the top left corner of the image and a row-column coordinate system).

1. Compute the connected components in the image.

2. Compute the upper-left (a_i), upper-right (b_i), lower-right (c_i), and lower-left (d_i) corners of the bounding box of each connected component.

3. Find the four feature points using the following equations:

$$\begin{aligned} p_1 &= \arg\min_{a_i}(x(a_i) + y(a_i)), \\ p_2 &= \arg\max_{b_i}(x(b_i) - y(b_i)), \\ p_3 &= \arg\max_{c_i}(x(c_i) + y(c_i)), \\ p_4 &= \arg\min_{d_i}(x(d_i) - y(d_i)). \end{aligned}$$

   The above equations compute the upper-left (p_1), upper-right (p_2), lower-right (p_3), and lower-left (p_4) feature points on the ideal image.

The above algorithm is also used to compute the corresponding four feature points $q_1, \ldots, q_4$ on the real image. Since sometimes noise blobs can appear in a real image, we check to see that the bounding box sizes of the components are within a specified tolerance. Furthermore, a potential problem can arise when two bounding boxes have their corners on a 45 degree line. A transformation T can be estimated using the corresponding points $p_1, \ldots, p_4$ and $q_1, \ldots, q_4$ by the methods described in section 5.

7 Registration results on scanned images

Unfortunately, none of the three geometric transformations described in sections 4 and 5 model the transformation very accurately. That is, the real points are displaced from ideal transformed points in some nonlinear fashion.

To investigate further, we drew a rectangle on a blank image, and then print and scan it. The opposite sides of the scanned figure were no longer equal in length. In fact, all the four sides were of different lenght, suggesting a projective transform, which could arise because the image plane and the original document plane are not parallel to each other. If the projective transform cannot model the transformation, the mismatch must arise from nonlinearities in the optical and mechanical systems. Note that these nonlinearities could be either in the printer or the photocopier or the scanner or in any combination of the three units. We suspect that the non-linearities in the sensor-motion accounts for most of the mismatch. Small perturbations around the nominal position could be cause of spatial quantization.

In figure 1 we show a subimage of a scanned image with the groundtruth (character bounding boxes) overlaid. We see that there is a lot of error. This error is not systematic over the entire page.

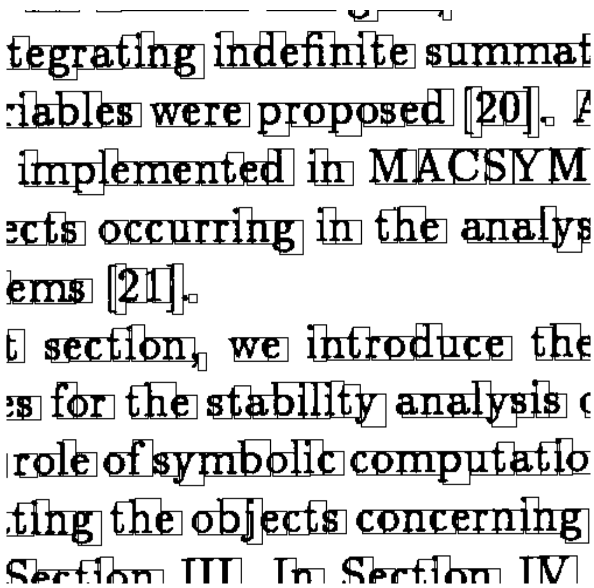


Figure 1: A scanned image with groundtruth overlaid. In this case perspective transform was used to register the ideal document image to the scanned image. It can be seen that there is large error in groundtruth

To confirm the fact that there are nonlinearities in the printing-photocopying-scanning processes, we set up a calibration experiment and performed a statistical test to prove the point that the similarity, affine and projective transforms alone do not model the transformation correctly. The calibration experiment is described in the following subsections.

7.1 The calibration experiment

We now describe a controlled experiment that was conducted to confirm the fact that the geometric transformation that occurs while printing and scanning documents is not similarity or affine or perspective. First we create an ideal calibration image consisting of only '+' symbols arranged in a grid. We print this document and then scan it back. The the crosses in the ideal image are then matched to the crosses in the scanned image. This set of corresponding points are then use to estimate the geometric transform parameters. The sample mean and sample covariance matrix of the registration error vectors are then computed. Since the population mean and population covariance matrix of the error vectors can be theoretically derived, we can test whether the theoretically derived distribution parameters are close to the experimentally gathered sample parameters.

In the next subsection we provide the details of the calibration data gathering process. In the subsequent subsection we give the details of the statistical hypothesis testing procedure.

7.2 Protocol for Calibration Experiment

The ideal image for calibrating the printer-photocopier-scanner process is created as follows. First a grid of equally spaced "+" symbols is arranged on a 3300×2500 binary image. The vertical and horizontal bars of the "+" symbol are 25 pixels long and 3 pixels thick. The number of symbols on each row and column of the grid are 23 and 30, respectively.

The ideal image is then printed and scanned. The intersection points of the two bars of the "+" symbols are used as the calibration points. The calibration points are detected by a morphological algorithm: first the image is closed with a 3×3 square structuring element. Next, two images are created by opening the closed image with a vertical and horizontal structuring elements, respectively. Calibration points on the scanned image are detected by binary-anding these two images. A connected component algorithm is then run on the image with the detected calibration points. The centroids of the connected components are used as the coordinates of the calibration points. The calibration points in the ideal image are known since the ideal calibration image is created under experimenter's control.

To estimate the projective transform, four feature points are first detected using the algorithm described in section 6. Next, we estimate the projective transform parameters from the ideal and real points (correspondences are known since we order the four points in a counter clockwise order, starting with the upper left feature point, and assume that the orientation of the page is unchanged). The estimated transform parameters are then used to project all the ideal points. An exhaustive search is conducted to establish correspondences between the projected ideal calibration points and real calibration points. That is, for each projected ideal point, we find the closest real point, and assume the two points match. This is done by a brute-force $O(n^2)$ algorithm. Since n is of the order of 1000, the computation required is of the order 10^6, which takes approximately three seconds on a Sparc 2. Next, for each calibration point we compute the registration error vector, which is the displacement vector between the real calibration point and the projected calibration point. The maximum error we attain is with ± 4 pixels in each coordinate.

In Figure 2(a) we show a subimage of the scanned calibration document. The detected calibration points are shown in Figure 2(b). In Figure 2(c) the ideal calibration points are transformed using the estimated projective transformation and overlaid on the real calibration points. A scatter plot of the error vectors is shown in Figure 3.

7.3 Statistical tests

Since the estimated parameters of the models are functions of real point coordinates, which are random variables, the estimated parameters are random variables. The distribution of estimated parameters can be derived in terms of the assumed distribution of the noise in the real point coordinates. To confirm that the geometric transformation model and the noise model are valid, we test whether or not the theoretically derived distribution of the estimated parameter vector is the same as that computed empirically. If either the geometric transformation model or the noise model is incorrect, the test for equality of the empirically computed distribution and the theoretically derived distribution will

(a) (b) (c)

Figure 2: (a) A subimage of the scanned calibration document. The detected calibration points are shown in (b). (c) The ideal calibration points are transformed using the estimated projective transformation and overlaid on the real calibration points.

not pass. Furthermore, instead of testing the distribution of the estimated parameters, we can test the distribution of the residual error, which in turn has a known distribution. A brief description of the hypothesis testing procedure is given in appendix A and the theory and software is described in detail in [Kan96, KH95].

8 Dealing with nonlinearities

As we saw, because of the nonlinearities, the groundtruth bounding boxes for the characters in a scanned image are not correct. Our solution to this problem is very simple. We first transform the ideal document image using the perspective transformation. The groundtruth associated with the ideal image is also transformed using the estimated perspective transform parameters. Next, each character in the perspective transformed image is locally translated and matched (using Hamming distance) with the a same size subimage in the scanned document image. Thus, if the the nonlinearity gave rise to a $(2, 3)$ translation error in pixels, our template matching process would give the best match (minimum Hamming distance) when the translation is $(2, 3)$. The size of the search window is decided by the calibration experiment. If the error vectors are large, the search window has to be made large. This local search process gives us a highly accurate groundtruth, and the potential errors are within a pixel.

9 Dealing with outliers: robust regression

At times, when two very similar characters (for example two 'i's, or one 'i' and one 'l') are close to each other, the template matching process can match the perspective transformed character to the wrong scanned character. This typically happens if we use a large search window size. This means that the error translation vector associated the wrongly matched character will be off. Fortunately, we have another way of detecting such outliers. Briefly the procedure is as follows. Once the error vectors are computed, we

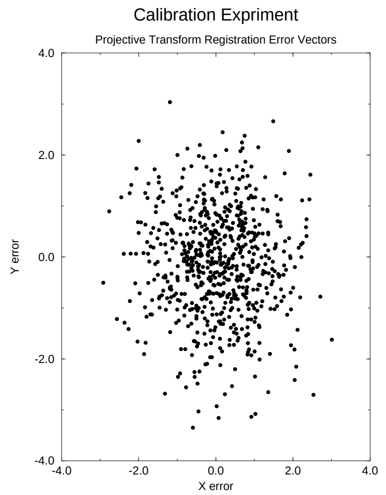


Figure 3: A scatter plot of the error vectors computed between real calibration points and projected ideal calibration points. The dimension is in pixels for each axis.

can fit a multivariate function to the x and y translation errors associated with characters in a small area of the image. It can be assumed that within this small area the error vectors will not vary much. Thus if we perform a robust regression, the outlier error vectors will be immediately detected, and can be corrected. For regression we use piecewise bilinear functions, which we now describe. A discussion on bilinear function fitting and image warping can be found in [Wol90].

Let the function $f : R^2 \to R^2$ be an image-to-image nonlinear function. We are given the ideal calibration points $p_1, \dots, p_n$, and the corresponding observed points $q_1, \dots, q_n$. That is, $q_i = f(p_i) + \eta$. The problem is to construct a piecewise bilinear function that approximates f in the sense that

$$\sum_{k=1}^{n} ||q_k - f(p_k)|| \tag{19}$$

is minimized.

The piecewise bilinear function is represented as follows. First, a grid of points $g_{i,j}$, with $i = 1, \dots, l$ and $j = 1, \dots, m$ on the first image are identified. The grid points are such that the y-coordinates of the points along any row of grid points are the same and the x-coordinates of points along any column of grid points are the same. That is, $y(g_{i,j}) = y(g_{i,k})$ for $j = 1, \dots, m$, $k = 1, \dots, l$. Furthermore, there is a natural ordering of the grid point coordinates: $x(g_{i,j}) < x(g_{i+1,j})$ and $y(g_{i,j}) < y(g_{i,j+1})$. Note that the number of grid points is much less than the number of calibration points: $l \times m < n$.

We represent the nonlinear function f by representing the transformation on the grid of points $g_{i,j}$. Let $g_{i,j} + \Delta g_{i,j}$ be the grid point after the function f transforms the grid point $g_{i,j}$. Let the point p lie within a grid cell whose four corner grid points are: $a = g_{i,j}, b = g_{i+1,j}, c = g_{i+1,j+1}, d = g_{i,j+1}$. The transformation of the point p is then approximated as follows. Let

$$t = (x(p) - x(a))/(x(b) - x(a)), \tag{20}$$

$$s = (y(p) - y(d))/(y(d) - y(a)). \tag{21}$$

Then the point $q = f(p) + \eta$ after transformation is given by

$$q = p + (1-t)(1-s)\Delta a + t(1-s)\Delta b + ts\Delta c + (1-t)s\Delta d + \eta, \tag{22}$$

where $\Delta a = \Delta g_{i,j}$; and $\Delta b, \Delta c$, and Δd are defined similarly.

Let a_k, b_k, c_k, d_k be the corner points of the grid cell within which the point p_k lies, and let t_k and s_k be constants calculated using equations (20) and (21). Equation (19) can be stated as: Find Δa_k, Δb_k, Δc_k, Δd_k to minimize

$$\sum_{k=1}^{n} ||q_k - [p_k + (1-t_k)(1-s_k)\Delta a_k + t_k(1-s_k)\Delta b_k + t_k s_k \Delta c_k + (1-t_k)s_k \Delta d_k]|| \;. \tag{23}$$

In the above equation, out of the $n \times 4$ elements Δa_k, Δb_k, Δc_k, Δd_k, $k = 1, \ldots, n$, only $l \times m$ elements are unique. For example, Δc_9 and Δd_{20} both might represent the same grid point variation, $\Delta g_{4,5} : \Delta c_9 = \Delta d_{20} = \Delta g_{4,5}$. We can now give unique labels to the grid differences, setup a system of linear equations, and solve for the unique elements in a least squares sense.

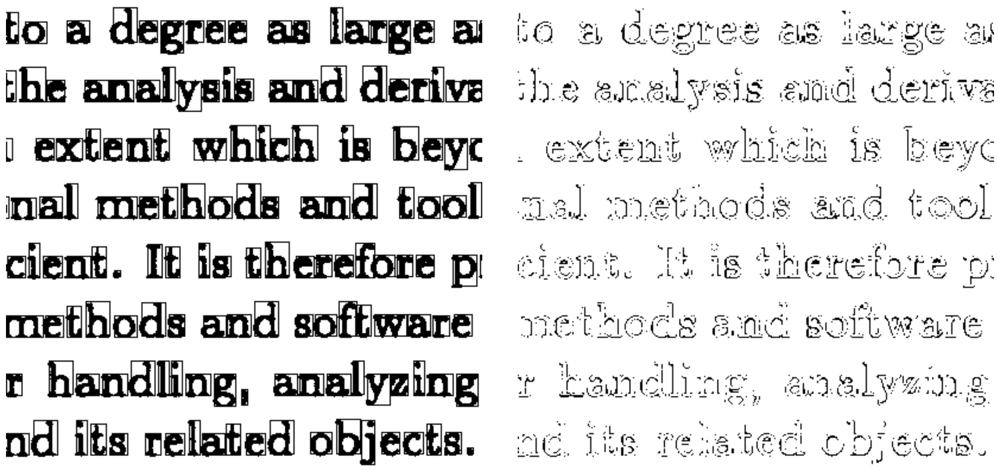
10 Experimental Protocol and Results

10.1 Data Collection

The ideal data is a LaTeX formatted document [Knu88, Lam86]. The IEEE Transaction style is used for typesetting the document for English documents. Hindi documents in Devanagari fonts are formatted using public domain LaTeX macros [Vel]. The ideal binary image and character ground truth is created using the DVI2TIFF software. The ideal document is created at 300×300 dots/inch resolution and the size of the binary document in pixels is 3300×2550. This document is printed using a SparcPrinter II. Next, the original printed document is photocopied five times using a Xerox photocopier - once at the normal setting, twice with darker settings, and twice with lighter settings. Finally the five photocopied documents are scanned using a Ricoh scanner. The scanner is set at 300×300 dots/inch resolution. The rest of the scanner parameters are set at normal settings. The scanned binary image is of size 3307×2544.

10.2 Protocol for Generating Real Ground Truth

Once the real scanned documents have been gathered as described in the previous section, we use the registration algorithm, described in section 1 to i) transform the ideal binary documents so that it registers to the scanned document and ii) to create the ground truth corresponding to the scanned document. The transformed ground truth also forms the ground truth for the transformed ideal document. The local nonlinearities of the transformation are accounted for by searching in a local neighborhood for a good match between the ideal character symbol and the real character symbol. The local template match window size is determined by the calibration experiment we performed earlier. Since the maximum error in the registration is ± 4 pixels, we used a window with $-7 \leq \Delta x, \Delta y \leq 7$. The ground truth generated by our algorithm is highly accurate. A subimage of the scanned image with the overlaid bounding box is shown in Figure 4. An exclusive or-ed image of the real scanned document and the registered ideal document is show in Figure 4. The exclusive OR images shows that the groundtruth for each character is centered on the character and the differences are at the character edge. These differences due to the image point spread function of the printing and scanning are what is expected. The time taken for this procedure on a SUN SPARC 5, is 2 minutes.

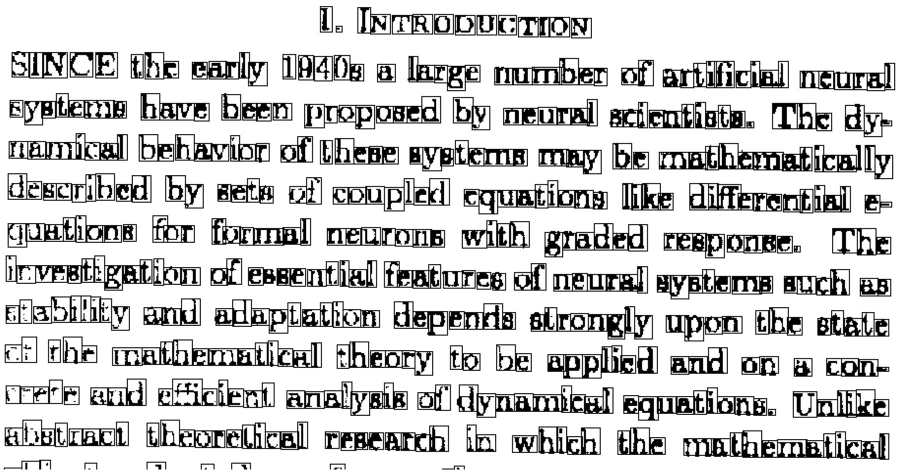


(a) (b)

Figure 4: Ground truth for real documents. (a) shows a subimage of a document with the estimated bounding boxes of each character. (b) shows the result of exclusive-OR between the real document and the registered ideal document. The exclusive OR images shows that the groundtruth for each character is centered on the character and the differences are at the character edge. These differences due to the image point spread function of the printing and scanning are what is expected.

11 Summary

In this paper we presented a closed-loop method for producing character groundtruth for real document images. The method starts by generating ideal noise-free document images using a document typesetting software like LaTeX. These binary document images are printed, photocopied/FAXed, and then scanned. Feature points are extracted from the ideal and the scanned document images, and their correspondences established. We showed that the similarity, affine and projective transformations alone cannot be used to represent the transformation between the ideal and the scanned documents. This fact was confirmed by using test images specially designed for calibration, and verifying that the statistical distribution of the registration error is not what the theory predicts. The local nonlinearities that exist can be accounted for by performing a local template match using the ideal character as the template, and searching a small neighborhood in the real image for the best match. The size of the local search neighborhood is decided by the calibration experiment. The calibration experiment gives us the maximum deviations that can occur between the ideal feature points after they have been transformed using



I. INTRODUCTION

SINCE the early 1940s a large number of artificial neural systems have been proposed by neural scientists. The dynamical behavior of these systems may be mathematically described by sets of coupled equations like differential equations for formal neurons with graded response. The investigation of essential features of neural systems such as stability and adaptation depends strongly upon the state of the mathematical theory to be applied and on a concrete and efficient analysis of dynamical equations. Unlike abstract theoretical research in which the mathematical

Figure 5: A subimage of a FAXed document with the groundtruth overlaid. Notice that the characters in the bottom left of the image are hardly legible. Maunal groundtruth for these type of documents would be prone to errors. In contrast, our software has produced correct groundtruth without any error.

the estimated transformation and the feature points on the scanned image. If character, word or zone groundtruth is available for any real document, the registration algorithm can be used to generate the corresponding groundtruth for a rescanned version of the document.

Keywords: Automatic real groundtruth, document analysis, performance evaluation, registration, geometric transformations.

Acknowledgements

Kanungo would like to thank Bhabuthosh Chanda, Ken Thornton, Harpreet Sawhney and B. K. P. Horn for comments.

References

[And84] T. W. Anderson. *Introduction to Multivariate Statistical Analysis*. John Wiley & Sons, Inc., New York, NY, 1984.

[CF90] R. G. Casey and D. R. Ferguson. Intelligent forms processing. *IBM Systems Journal*, 29(3):435–50, 1990.

[DR93] D. S. Doermann and A. Rosenfeld. The processing of form documents. In *Proc. of Int. Conf. on Document Analysis and Recognition*, pages 497–501, Tsukuba, Japan, October 1993.

[HP+94] R. M. Haralick, I. Phillips, et al. U.W. English Database I, 1994.

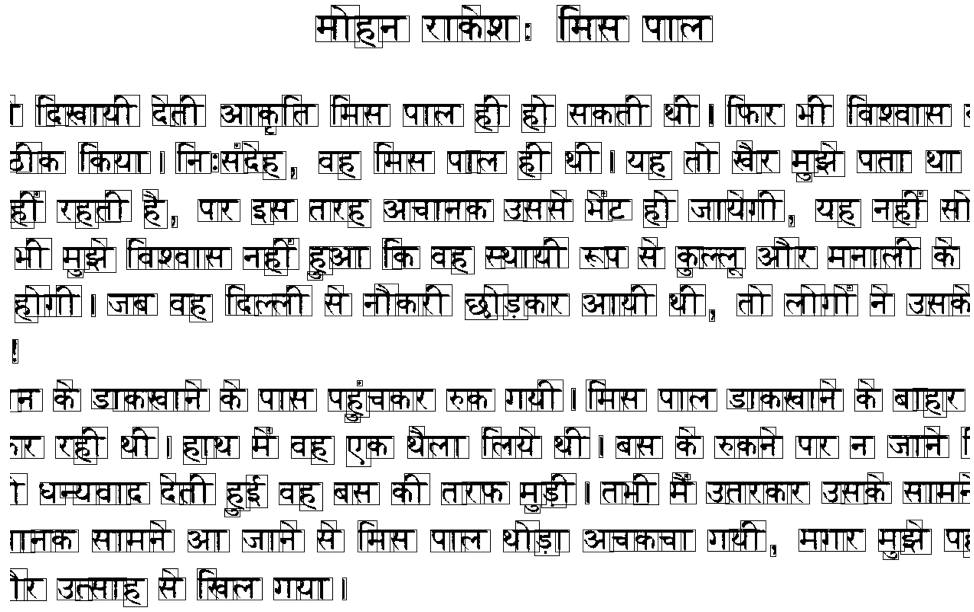


Figure 6: A subimage of a LaTeX formatted Hindi document in Devanagri script with the groundtruth overlaid. The bounding boxes of the symbols are overlapping because the Devanagri symbols are arranged in that manner.

[Kan96] T. Kanungo. *Document Degradations Models and a Methodology for Degradation Model Validation.* PhD thesis, University of Washington, Seattle, WA., 1996.

[KBH95] T. Kanungo, H. S. Baird, and R. M. Haralick. Validation and estimation of document degradation models. In *Proc. of Fourth Annual Symposium on Document Analysis and Information Retrieval*, Las Vegas, Nevada, April 1995.

[KH95] T. Kanungo and R. M. Haralick. Multivariate hypothesis testing for gaussian data: Theory and software. Technical Report ISL Tech. Report: ISL-TR-95-05, University of Washington, Dept. of Electrical Engineering, University of Washington, Seattle, WA 98195, October 1995.

[KH96] T. Kanungo and R. M. Haralick. Automatic generation of character groundtruth for scanned documents: A closed loop approach. In *Proc. of IAPR International Conference on Pattern Recognition*, pages 669–675, Vienna, Austria, August 1996.

[KHB+94] T. Kanungo, R. M. Haralick, H. S. Baird, W. Stuetzle, and D. Madigan. Document degradation models: Parameter estimation and model validation. In *Proc. of Int. Workshop on Machine Vision Applications*, Kawasaki, Japan, December 1994.

[KHB95] T. Kanungo, R. M. Haralick, and H. S. Baird. Power functions and their use in document degradation model validation. In *Proc. of Second International Conference on Document Analysis and Recognition*, Montreal, Canada, August 1995.

[KHP93] T. Kanungo, R. M. Haralick, and I. Phillips. Global and local document degradation models. In *Proc. of Second International Conference on Document Analysis and Recognition*, pages 730–734, Tsukuba, Japan, October 1993.

[KHP94] T. Kanungo, R. M. Haralick, and I. Phillips. Non-linear local and global document degradation models. *Int. Journal of Imaging Systems and Technology*, 5(4), 1994.

[Knu88] D. E. Knuth. *TEX: the program.* Addison-Wesley, Reading, Mass., 1988.

[Lam86] L. Lamport. *LATEX: a document preparation system.* Addison-Wesley, Reading, Mass., 1986.

[Vel] F. J. Velthuis. Devanagri macro for Latex. FTP: cs.ducke.edu/dist/sources/devanagari.tar.Z.

[Wol90] G. Wolberg. *Digital Image Warping.* IEEE Computer Society Press, Los Alamitos, CA, 1990.

A Statistical testing of distribution parameters

In this appendix we give a test procedure for testing the null hypothesis that the population mean vector, μ, and population covariance, Σ,are equal to a particular vector, μ_0, and matrix, Σ_0, respectively. The test statistic is based on the sample mean and sample covariance matrix, and its null distribution is a χ^2 distribution. For a detailed discussion see [KH95, And84].

The null hypothesis H_N is:

$H_N : \Sigma = \Sigma_0$, and $\mu = \mu_0$.

Let $\bar{x}$ and S be defined as:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

and

$$S = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^t,$$

where we have assumed that the data vectors x_i are p-dimensional and the sample size is n. Let $B = (n-1)S$ and

$$\lambda = (e/n)^{pn/2}|B\Sigma_0^{-1}|^{n/2}\exp\left(-[tr(B\Sigma_0^{-1}) + n(\bar{x} - \mu_0)^t\Sigma_0^{-1}(\bar{x} - \mu_0)]/2\right).$$

Then the test statistic, T, is:

$$T = -2\log\lambda \tag{24}$$

Distribution of the test statistic T under true null hypothesis is χ^2 (for proofs see [And84]):

$$T \sim \chi^2_{p(p+1)/2+p}.$$

Thus the test procedure with a significance level α is:

1. Compute the test statistic t.

2. Compute the p-value:

$$pvalue = \mathrm{Prob}(\ T > t | T \sim \chi^2_{p(p+1)/2+p}).$$

3. If $pvalue < \alpha$, reject the null hypothesis, H_N.