Pattern Recognition Letters 32 (2011) 1912–1921

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Statistical mixture model for documents skew angle estimation ☆

Amir Egozi *, Its'hak Dinstein

Ben-Gurion University of the Negev, Electrical and Computer Engineering Department, Beer-Sheva 84105, Israel

ARTICLE INFO

Article history:
Received 23 June 2010
Available online 23 July 2011
Communicated by H. Sako

Keywords:
Skew detection
Statistical mixture models
Expectation Maximization (EM)
Linear least squares

ABSTRACT

We present a statistical approach to skew detection, where the distribution of textual features of document images is modeled as a mixture of straight lines in Gaussian noise. The Expectation Maximization (EM) algorithm is used to estimate the parameters of the statistical model and the estimated skew angle is extracted from the estimated parameters. Experiments demonstrate that our method is favorably comparable to other existing methods in terms of accuracy and efficiency.

1. Introduction

Digital images of documents may be rotated or skewed at an arbitrary angle because of the way it was scanned or because of the document feeder tolerance. In case of handwritten historical documents (Likforman-Sulem et al., 2007), the skew can be part of the handwriting inaccuracy. Document skew in many cases affects negatively the accuracy of character segmentation and recognition. Thus, automatic detection and correction of skew is a sought-after function (Hull, 1998).

Document skew can be either global, i.e. all text lines have the same orientation, or local where lines or part of lines can be rotated in different angles. Here we focus on global skew estimation, and assume that the skew is uniform within the text line.

1.1. Skew detection methods

Several classes of skew detection algorithms are considered in (Hull, 1998), among them are the projection profile technique (Bloomberg et al., 1995; Postl, 1986), analysis of the geometric distribution of text features (Baird, 1987; Chen and Haralick, 1994) and the Hough transform (Amin and Fischer, 2000; Duda and Hart, 1972; Srihari and Govindaraju, 1989). The general characteristics of each class is discussed and application examples are presented. An experimental evaluation of skew estimation algorithms is available in (Bagdanov and Kanai, 1996).

Other methods include, nearest neighbor clustering (Hashizume et al., 1986; Liolios et al., 2001; Smith, 1995), and analysis of the background image (Bar-Yosef et al., 2008; Lu et al., 2007).

A straightforward solution to determining the skew angle of document images uses a *horizontal projection profile*. This is a one-dimensional array with length equal to the number of pixel rows in the image. The direction of the projection should be perpendicular to the text lines. Since these directions are not known, a number of projection profiles must be calculated. Each element in a projection profile array stores a count of the number of text pixels in the corresponding direction. This histogram has the maximum amplitude and frequency when the text in the image is not skewed.

The projection profile method is simple and well understandable, but the range of detectable angles is restricted because the profile computation for many angles is a time consuming operation. Bloomberg et al. (1995) improved the basic approach by downsampling the image before calculating the projection profile (to reduce computational cost). This is done in a way that preserves the horizontal structure in the image. Moreover, a search algorithm was used that first calculate the projection profiles over a sequence of angles that have a coarse resolution. The angle that maximizes a criterion function is used as the center for a finer resolution search for the skew angle.

Another class of techniques for document image skew detection reduces the computational complexity by first extracting the (x,y)-coordinates of some feature points in an image. All subsequent computations are performed on those coordinates and the image itself is not accessed after the feature extraction stage.

One method of feature extraction that has been used is based on first locating the connected component (CC) in the image. These are the groups of connected pixels belonging to objects in the

☆ An early and short version of this work appeared in the preceding of ICDAR'07 (Egozi and Dinstein, 2007).

* Corresponding author.

E-mail addresses: agozi@ee.bgu.ac.il (A. Egozi), dinstein@ee.bgu.ac.il (I. Dinstein).

0167-8655/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2011.07.004

image. The (x,y)-coordinates of one representative point for each component (e.g. the centroid) can be used to determine the skew angle by analyzing their projection profiles in the same way as done with all the text pixels.

Baird (1987) proposed a method similar to that discussed above in which the feature points are the bottom-centers of each connected component. Since a brute-force implementation of this approach would be computationally prohibitive to achieve the desired accuracy, the author presents a clever approach that uses successive approximation at finer resolutions.

In (Chen and Haralick, 1994) the image is preprocessed with recursive morphological transforms. They are designed to remove the ascenders, descenders, and overfills from each character. The desired result is one connected component for each text line. A least square procedure is used to fit a line to each connected component. A histogram is constructed of the angles of the detected lines. The skew angle of the document is determine from the histogram using a search procedure. This is necessary since there may be significant disagreement among the angles of the fitted lines.

The Hough transform is a popular approach to document image skew detection (Srihari and Govindaraju, 1989; Amin and Fischer, 2000). This approach used the fact that the highest number of co-linear pixels are on lines that are co-incident with the baseline of the text. This is similar to the characteristic exploited by the projection profile methods. Variants of this basic algorithm includes – different selection of feature points (e.g. centroids/bottom of connected components, edges), down-sampling to reduce computations cost and using hierarchical approaches.

Skew detection by clustering of textual components exploits the general assumption that characters in a line are aligned and close to each other. Hashizume et al. (1986) present a bottom up technique based on nearest neighbor clustering. For each component they compute the direction of the segment that connect it to its geometrically nearest neighbor. These direction are accumulated in a histogram whose maximum provides the dominant skew angle.

The method described by Smith (1995) is based on the clustering of the connected components into text lines. The components are grouped into lines as follows: for each component the degree of vertical overlap with existing lines, if any, is computed. The current component is assigned to a new line or to an existing one, depending on its degree of vertical overlap. For each cluster the baseline skew is estimated by means of a least median of squares fit. The global page skew is computed as the median slope.

Recently, several works based their skew detection algorithm on the analysis of the background of document images. This approach is based on the assumption that text images normally hold a large amount of equal distant interline spacings. Lu et al. (2007) analyzed the horizontal and vertical white-runs histograms of the background in order to determine the skew angle. Bar-Yosef et al. based their skew detection algorithm on the background distance transform (DT), where each pixel is represented by its shortest Euclidean distance to a text component. The skew angle is extracted from the histogram of the DT's gradient orientations.

The least squares method has been extensively used in various skew detection algorithms for fitting a straight line to a set of feature points. However, these algorithms (Cao et al., 2003; Chen and Haralick, 1994; Liolios et al., 2001; Yu et al., 1995) have an intrinsic drawback, which is the need to group feature points into line representative groups. In our algorithm, this is not required since the algorithm estimates the parameters of multiple lines that fit the textual feature points simultaneously. The feature points can be any feature of the connected components in the image, e.g. the centroid, or any other features that represent a line of text or characters.

1.2. The proposed EM based algorithm

Our proposed method is based on a statistical mixture model where each component represents a straight line corrupted by Gaussian noise. The estimation of the model parameters is obtained using maximum likelihood estimation by the Expectation Maximization (EM) algorithm (Dempster et al., 1977). The skew angle estimate is extracted from the histogram of the slope angles of the estimated lines. The algorithm is easy to implement and it is efficient since only simple operations are needed.

Our method can estimate an arbitrary skew angle and it can also detect the skew angle in documents including graphics or pictures in a moderate quantity. Experimental results show that the algorithm is adequate for printed scanned documents as well as for unconstrained handwritten documents, particularly historical documents.

This paper is organized as follows: in Section 2 we describe the general statistical mixture models and the EM algorithm. In Section 3 we introduce our mixture model and the EM equations used for skew detection. Section 4 describes the experimental results and Section 5 concludes the paper.

2. EM algorithm for mixture models

2.1. Statistical mixture models – introduction and notation

In a mixture model, a probability density function is expressed as a linear combination of basis functions. A model with M components is written in the form

$$p(y) = \sum_{j=1}^{M} \pi_j p(y|\theta_j), \tag{1}$$

where $p(y|\theta_j)$ is a given family of densities with a parameter vector (or scalar) θ_j, that typically varies with j. We call these functions *component densities*. The *mixing coefficients* or *weights* – π_j, are the probabilities of choosing component j out of $M, \pi_j = p(j), j = 1, \ldots, M$. The weights satisfy – $0 \leqslant \pi_j \leqslant 1$ and $\sum_{j=1}^{M} \pi_j = 1$. These constraints guarantee that the model represents a valid density function.

2.2. Maximum likelihood (ML) estimation by the EM algorithm

A common way for determining the parameters of the mixture model from an observed data set is based on maximizing the data likelihood $\mathcal{L}(\boldsymbol{\theta}) = p(\mathbf{y}|\boldsymbol{\theta})$, where $\mathbf{y} = [y_0 y_2, \ldots, y_{N-1}]^T$ consists of N statistically independent observations and $\boldsymbol{\theta}$ is the set of all unknown parameters. In the case of mixture models, $p(\mathbf{y}|\boldsymbol{\theta})$ has the form of (1) and $\boldsymbol{\theta} = \{\theta_1, \theta_2, \ldots, \theta_M, \pi_1, \pi_2, \ldots, \pi_M\}$.

The maximum likelihood estimate is defined as

$$\begin{aligned}\hat{\boldsymbol{\theta}}_{ML} &= \arg\max_{\boldsymbol{\theta}} p(\mathbf{y}|\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} \log p(\mathbf{y}|\boldsymbol{\theta}) \\ &= \arg\max_{\boldsymbol{\theta}} \sum_{n=0}^{N-1} \log p(y_n|\boldsymbol{\theta}).\end{aligned} \tag{2}$$

The *Expectation Maximization* (EM) algorithm (Dempster et al., 1977) is a general method of finding the maximum-likelihood estimate of the parameters of an underlying distribution from a given data set when the data is incomplete or has missing values. In the context of mixture models the missing information is the component that generates a specific data point. Since this information is unavailable, we consider a hypothetical complete data set (y_n, z_n) in which each data point is labeled with the component that generated it. For each data point y_n the corresponding random variable z_n can get an integer value in the range $1,\ldots,M$. However, since we do not know the distribution of the z_n, we adopt the following procedure. First, we guess some values for the parameters of the mixture model $\boldsymbol{\theta}^{(0)}$ and we use these, together with Bayes theorem, to find the distribution of $\mathbf{z}=\{z_n\}_{n=0}^{N-1}$. We then compute the expectation of the complete data log likelihood with respect to this distribution. This is the *expectation* or the *E*-step of the algorithm (see Eq. (3) below). In the *maximization* or *M*-step, we maximize the expectation result in order to find a new set of parameters (see Eq. (4) below). The algorithm is guaranteed to increase the likelihood at each step until a local maximum is found (Dempster et al., 1977).

Starting with an initial estimate of the parameters $\boldsymbol{\theta}^{(0)}$, The EM algorithm for general mixture models can be formulate as follows:

Expectation – calculate the function:

$$Q(\boldsymbol{\theta},\boldsymbol{\theta}^{(k)})=\sum_{n=1}^{N}\sum_{j=1}^{M}p\left(j|y_n,\boldsymbol{\theta}^{(k)}\right)\log\left\{\pi_j p(y_n|\theta_j)\right\}. \tag{3}$$

Maximization – optimize $Q(\boldsymbol{\theta},\boldsymbol{\theta}^{(k)})$ with respect to $\boldsymbol{\theta}$

$$\boldsymbol{\theta}^{(k+1)}=\arg\max_{\boldsymbol{\theta}} Q\left(\boldsymbol{\theta},\boldsymbol{\theta}^{(k)}\right). \tag{4}$$

3. EM for mixture of straight lines in Gaussian noise

3.1. Mixture model of straight lines in Gaussian noise

Our mixture model consists of multiple regression problems. In this scheme, we observe dependent variables $y_n, n=0,\ldots,N-1$, and explanatory variables $x_n, n=0,\ldots,N-1$, according to

$$y_n=a_j+b_jx_n+e_n \tag{5}$$

for $n=0,\ldots,N-1$ and $j=1,\ldots,M$, where e_n is an i.i.d Gaussian noise with zero mean and σ_j^2 variance. Thus, the parameters to be estimated are the slope b_j, the intercept a_j, the noise variance σ_j^2 and the weight π_j for all $j=1,\ldots,M$

$$\boldsymbol{\theta}=\{\pi_j,\boldsymbol{\theta}_j\}_{j=1}^{M}=\left\{\pi_j,\left(a_j,b_j,\sigma_j^2\right)\right\}_{j=1}^{M}. \tag{6}$$

Since the e_n distribution function is Gaussian, the full expression for the jth component distribution is

$$p(y_n|\boldsymbol{\theta}_j)=\frac{1}{\sqrt{2\pi\sigma_j^2}}\exp\left\{-\frac{(y_n-a_j-b_jx_n)^2}{2\sigma_j^2}\right\}, \tag{7}$$

which is the distribution of the vertical offsets of the feature points (x_n,y_n) from the line $y_n=a_j+b_jx_n$. This is a Gaussian distribution for y_n with mean $a_j+b_jx_n$ and variance σ_j^2, i.e. $y_n\sim\mathcal{N}\left(a_j+b_jx_n,\sigma_j^2\right)$. Thus, the mixture model of M lines can be formulated as

$$p(y_n|\boldsymbol{\theta})=\sum_{j=1}^{M}\pi_j\frac{1}{\sqrt{2\pi\sigma_j^2}}\exp\left\{-\frac{(y_n-a_j-b_jx_n)^2}{2\sigma_j^2}\right\}. \tag{8}$$

An example is shown in Fig. 1. A synthetic data set was generated from two lines $y=10+2x$ and $y=80-x$, which corresponds to the distribution $p(y_n)=0.5\mathcal{N}(10+2x_n,\sigma_1^2)+0.5\mathcal{N}(80-x_n,\sigma_2^2)$, where $\sigma_1=20$ and $\sigma_2=5$. The dotted lines are the true underlying lines, and the solid lines are the estimated lines using the proposed method.

In order to estimate the parameters of the model (8), we adopt the EM algorithm. Substitution (7) into (3) gives

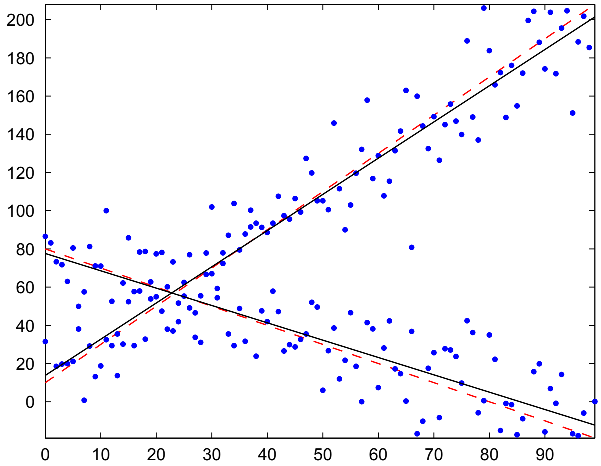


Fig. 1. A set of 200 points generated from the mixture model $p(y_n)=0.5\mathcal{N}(10+2x_n,\sigma_1^2)+0.5\mathcal{N}(80-x_n,\sigma_2^2)$, where $\sigma_1=20$ and $\sigma_2=5$. The dotted lines are the true underlying lines, and the solid lines are the estimated lines using the proposed method.

$$Q(\boldsymbol{\theta},\boldsymbol{\theta}^{(k)})=\sum_{n=0}^{N-1}\sum_{j=1}^{M}\alpha_{jn}^{(k)}\left\{\log\pi_j-\log\sigma_j-\frac{(y_n-a_j-b_jx_n)^2}{2\sigma_j^2}\right\}, \tag{9}$$

where $\alpha_{jn}^{(k)}$ is the posterior distribution of the component labels given the observed data and the previous-step parameters.

The full expression for $\alpha_{jn}^{(k)}$ according to Bayes' rule is

$$\begin{aligned}\alpha_{jn}^{(k)}&=p\left(j|y_n,\pi_j^{(k)},a_j^{(k)},b_j^{(k)},\left(\sigma_j^{(k)}\right)^2\right)\\&=\frac{\pi_j^{(k)}p\left(y_n|a_j^{(k)},b_j^{(k)},\left(\sigma_j^{(k)}\right)^2\right)}{\sum_{j=1}^{M}\pi_j^{(k)}p\left(y_n|a_j^{(k)},b_j^{(k)},\left(\sigma_j^{(k)}\right)^2\right)}.\end{aligned}$$

The $\alpha_{jn}^{(k)}$'s are calculated using the previous-step parameters, therefore, they are constant with respect to the *Maximization* step.

The maximization of (9) with respect to a_j, b_j and σ_j is straightforward using partial derivatives with respect to each unknown parameter. However, for the mixing coefficients $\{\pi_j\}_{j=1}^{M}$ we must take into account the constraint $\sum_{j=1}^{M}\pi_j=1$. This is done by introducing a Lagrange multiplier λ and maximizing the function

$$\widetilde{Q}=Q+\lambda\left(\sum_{j=1}^{M}\pi_j-1\right), \tag{10}$$

setting the derivatives of (10) with respect to π_j to zero and using the constraint, we obtain the following update equation for the mixing coefficient

$$\pi_j^{(k+1)}=\frac{1}{N}\sum_{n=0}^{N-1}\alpha_{jn}^{(k)}. \tag{11}$$

This is a general result for mixture models. The derivatives with respect to a_j and b_j result in two linear equations. These two equations can be presented in matrix notation:

$$\begin{bmatrix}\sum_{n=0}^{N-1}\alpha_{jn}^{(k)} & \sum_{n=0}^{N-1}\alpha_{jn}^{(k)}x_n\\ \sum_{n=0}^{N-1}\alpha_{jn}^{(k)}x_n & \sum_{n=0}^{N-1}\alpha_{jn}^{(k)}x_n^2\end{bmatrix}\begin{bmatrix}a_j^{(k+1)}\\ b_j^{(k+1)}\end{bmatrix}=\begin{bmatrix}\sum_{n=0}^{N-1}\alpha_{jn}^{(k)}y_n\\ \sum_{n=0}^{N-1}\alpha_{jn}^{(k)}x_ny_n\end{bmatrix}, \tag{12}$$

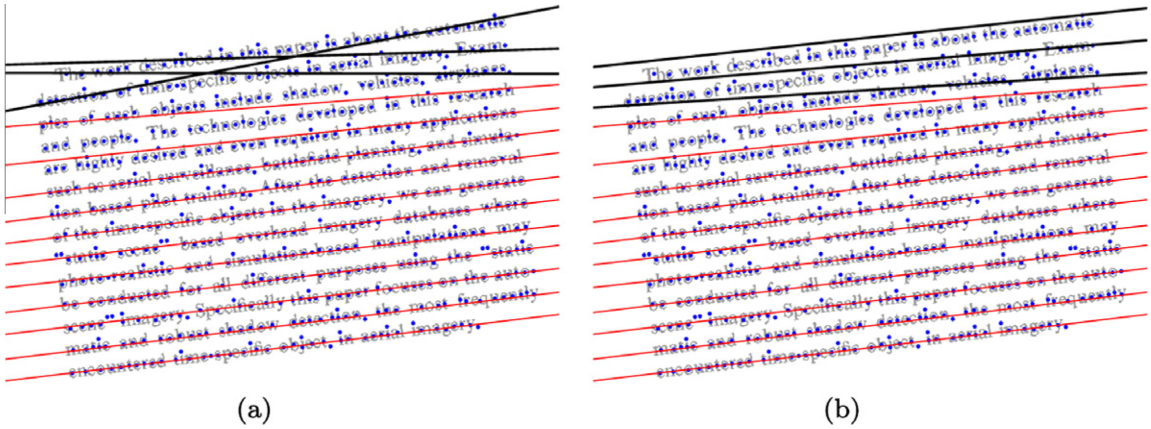


Fig. 2. An example of the correction of intersecting lines. See text for details.

which is a modified version of the regular linear-least squares equations,[1] with $\alpha_{jn}^{(k)}$ as weights for each observation.

The update equations for the parameters $a_j^{(k+1)}$ and $b_j^{(k+1)}$ require the inversion of the square matrix in (12), and multiplication of both sides of the equation with the inverted matrix, which yield

$$a_j^{(k+1)} = \frac{\sum_{n=0}^{N-1}\alpha_{jn}^{(k)} y_n \sum_{n=0}^{N-1}\alpha_{jn}^{(k)} x_n^2 - \sum_{n=0}^{N-1}\alpha_{jn}^{(k)} x_n \sum_{n=0}^{N-1}\alpha_{jn}^{(k)} x_n y_n}{N\sum_{n=0}^{N-1}\alpha_{jn}^{(k)} x_n^2 - \left(\sum_{n=0}^{N-1}\alpha_{jn}^{(k)} x_n\right)^2},$$
$$b_j^{(k+1)} = \frac{N\sum_{n=0}^{N-1}\alpha_{jn}^{(k)} x_n y_n - \sum_{n=0}^{N-1}\alpha_{jn}^{(k)} x_n \sum_{n=0}^{N-1}\alpha_{jn}^{(k)} y_n}{N\sum_{n=0}^{N-1}\alpha_{jn}^{(k)} x_n^2 - \left(\sum_{n=0}^{N-1}\alpha_{jn}^{(k)} x_n\right)^2}. \qquad (13)$$

Given $a_j^{(k+1)}$ and $b_j^{(k+1)}$ the update equation for the variance is

$$\left(\sigma_j^{(k+1)}\right)^2 = \frac{\sum_{n=0}^{N-1}\alpha_{jn}^{(k)}\left(y_n - a_j^{(k+1)} - b_j^{(k+1)} x_n\right)^2}{\sum_{n=0}^{N-1}\alpha_{jn}^{(k)}}. \qquad (14)$$

3.2. EM equations for document skew detection

For document skew detection we have good prior knowledge of the structure of the data. We know that text lines are usually equally spaced, therefore it is reasonable to assume that the residuals in the regression formula (5) will have constant variance for all lines, i.e. σ_j^2 will equal σ^2 for all $j = 1, \ldots, M$. Calculating this variance is easily done by replacing σ_j with σ in (9). Differentiating Q with respect to σ^2 and setting the result to zero gives the update equation

$$\left(\sigma^{(k+1)}\right)^2 = \frac{\sum_{n=0}^{N-1}\sum_{j=1}^{M}\alpha_{jn}^{(k)}\left(y_n - a_j^{(k+1)} - b_j^{(k+1)} x_n\right)^2}{\sum_{n=0}^{N-1}\sum_{j=1}^{M}\alpha_{jn}^{(k)}}. \qquad (15)$$

This equation can be used in the EM algorithm to obtain an estimated model with constant variance components.

In addition, text lines are usually parallel, meaning that we can expect to obtain almost the same slope from all the estimated lines. Setting the slope $b_j = b$ for $j = 1, \ldots, M$ we follow the same procedure that we used to obtain (15) and arrive at the update equation for the slope

$$b^{(k+1)} = \frac{\sum_{n=0}^{N-1}\sum_{j=1}^{M}\alpha_{jn}^{(k)} x_n (y_n - a_j)}{\sum_{n=0}^{N-1}\sum_{j=1}^{M}\alpha_{jn}^{(k)} x_n^2}. \qquad (16)$$

Using this value for the slope, the intercepts $a_j^{(k)}$ are then calculated according to

$$a_j^{(k+1)} = \frac{\sum_{n=0}^{N-1}\alpha_{jn}^{(k)} y_n - b^{(k+1)}\sum_{n=0}^{N-1}\alpha_{jn}^{(k)} x_n}{\sum_{n=0}^{N-1}\alpha_{jn}^{(k)}}. \qquad (17)$$

Using (16) and (17) together with (15) yielded good results for printed text images and for small skew angles, namely −5° to 5°. However, for general skew angles and for handwritten documents we applied a different approach.

Instead of estimating only parallel lines using (16) and (17), in any iteration of the algorithm we check whether there are intersections between the model's components. This can be easily verified by examine the vertical order of the lines end points. Intersecting lines occur only in successive lines, therefore, we identify the end points of the intersecting lines set and arrange the lines uniformly between these points. An illustrated example is given in Fig. 2 for a 3 lines intersection.

When using the update Eqs. (16) and (17), the estimated skew angle is $\phi = \tan^{-1}(b)$. Otherwise, we extract the skew angle from the estimated slope angles of the mixture model $\{\tan^{-1}(b_j)\}_{j=1}^{M}$. This is done using the kernel density estimation (Parzen Windows) technique (Bishop, 1995). The slope angles, $\tan^{-1}(b_j)$, are considered as independent observations from a statistical distribution $\hat{p}(x; \{\tan^{-1}(b_j)\}_{j=1}^{M})$, which is estimated as a superposition of Gaussian kernels. Each kernel is centered at $\tan^{-1}(b_j)$ with bandwidth (variance) h, which is automatically detected using the method of Sheather and Jones (1991). The estimated skew angle ϕ is chosen as the maximum of the estimated distribution

$$\phi = \arg\max_x \hat{p}\left(x|\{\tan^{-1}(b_j)\}_{j=1}^{M}\right). \qquad (18)$$

This choice renders the skew angle estimate more robust to errors because the skew angle estimate is determined according to most of the model's estimated lines.

Finally, we wish to consider one implementation issue. One can easily verify that the model parameters for a line component can not be determined if $\sum_{n=0}^{N-1}\alpha_{jn} = 0$ for any j. This corresponds to a line component that is not associated to any feature point and has a prior probability equal to zero. Therefore, we check at each iteration if any of the line components has zero prior. If this occurs, the corresponding line component is deleted and the algorithm continues its iterations. Since the deleted line component had zero

[1] The linear least square equations are –

$$\begin{bmatrix} \sum_{n=1}^{N} x_n & N \\ \sum_{n=1}^{N} x_n^2 & \sum_{n=1}^{N} x_n \end{bmatrix} \begin{bmatrix} a_j \\ b_j \end{bmatrix} = \begin{bmatrix} \sum_{n=1}^{N} y_n \\ \sum_{n=1}^{N} x_n y_n \end{bmatrix}.$$

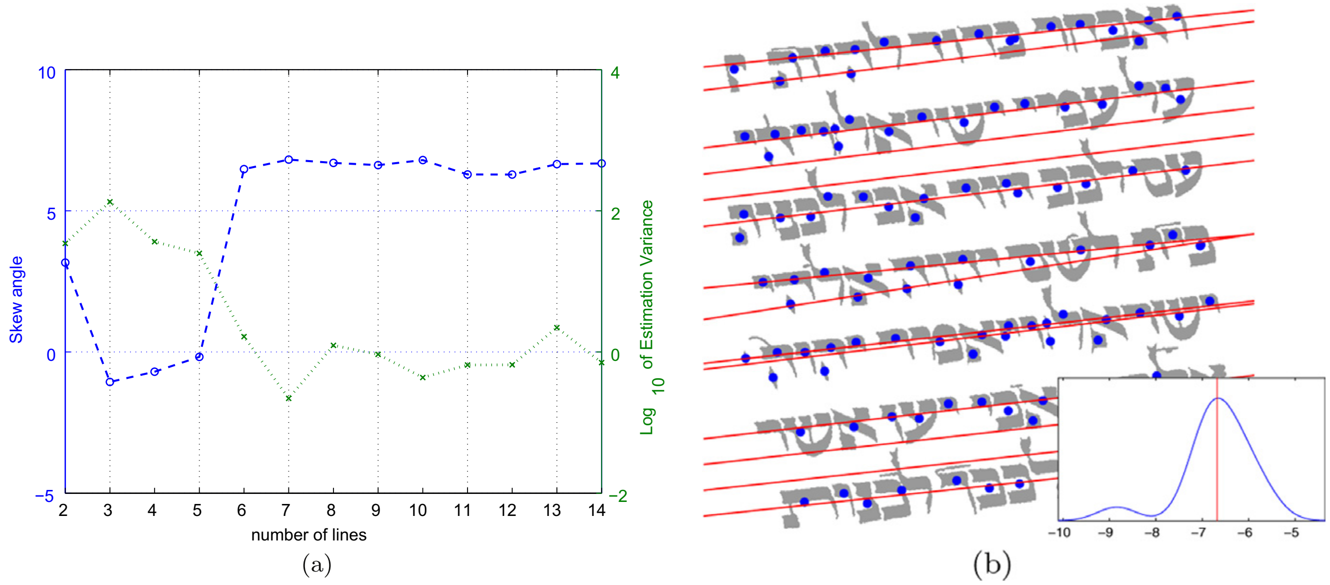


Fig. 3. Example of our model order selection criterion result. A document image containing STAM handwriting was rotated at 7° and the skew angle was estimated by mixture models with number of lines ranging from 2 to 14. (a) The variance of the skew angle estimation and the estimated skew angle. A strong correlation can be seen between the estimation accuracy and the estimation variance. The lowest estimation variance was achieved for the correct number of lines, namely 7; (b) even with more lines than needed, the skew angle is estimated quite accurately using kernel density estimation of the slope angles.

prior, and hence it was not responsible for any feature points, the rest of the line component parameters do not need to be changed.

3.3. Number of lines estimation

One of the problems in general mixture model parameter estimation is how to select the model order, i.e. the number of components, using the observed data. Classical approaches to the model order selection problem penalize the model fit according to a measure of their complexity. Examples of these methods are the AIC (Akaike, 1974) and the BIC (Schwarz, 1978) criteria. Each penalizes the estimated model log likelihood with a function of the number of free parameters of the model. In our case, however, using such methods as the AIC and BIC criteria will not work since the number of free parameters for each added component (line) is exactly 3, i.e. a_j, b_j, σ_j^2. Experiments have shown that this is not sufficient to penalize complex models and give a high score to the appropriate model.

Numerous researchers believe that it is unwise to separate the model selection process from the specific goal of inference (Bouchard and Celeux, 2006). Since we are interested in estimating the skew angle of text images, a natural choice would be to use the variance of the estimated skew angle for model order selection. Therefore, for a model selection criterion we use:

$$\widehat{M} = \arg\min_{m} \left\{ var(\hat{f}(b|\mathcal{M}_m)),\ m \in [L_{min}, L_{max}] \right\}, \tag{19}$$

where $\hat{f}(b|\mathcal{M}_m)$ is the estimated density given an m-lines model, $\mathcal{M}_m$, and its line slopes $b = \{b_j\}_{j=1}^{M}$. The estimated density is obtained using kernel density estimation as has been described before. The model from a set of candidate models $\{\mathcal{M}_m | m \in [L_{min}, L_{max}]\}$ that achieves the minimum estimation variance is selected.

Fig. 3 illustrates the results for the model selection criterion for an image of Hebrew calligraphic handwriting. The data for the estimation of the model parameters were the centroids of the connected components. The left subfigure shows the $\log_{10}$ of the variance of the estimated density as a function of the number of lines as well as the estimated skew angle from each model. As can be seen, the lowest estimation variance was obtained from the model with the correct number of lines (7). Note that the lowest estimation variance not always agrees with the actual number of text lines in the image. However, numerous experiments have shown that the lowest variance is obtained together with the most accurate skew angle.[2] Another observation is that all models with more than seven lines produced a good approximation of the skew angle. The right subfigure shows the estimated model with 14 lines superimposed on the document image as well as the estimated line slopes distribution. From the line slope distribution it can be observed that not all the lines need to be parallel in order to obtain a correct skew angle estimate. This feature makes our skew detection method more robust and more accurate.

According to the previous observation, we propose the following solution to the model selection problem. We use vertical cuts at random locations to obtain an initial over-estimated number of lines. Then a fine search is conducted around the initial line number to obtain a final model order according to the estimation variance (19).

3.4. Initialization

In all our experiments the initialization of the algorithm for skew detection was as follows. The model lines parameters corresponded to horizontally parallel lines equally distributed over the image. This means that the slopes $b_j = 0$ for all $j = 1, \ldots, M$ where M is the estimated number of lines,[3] and the intercepts had the same deviation from each other, i.e. $|a_{i+1} - a_i| = \Delta$ for all $i \leqslant M - 1$. The prior probabilities were initiated according to $\pi_j = 1/M$ and the variances according to $\sigma_j^2 = M\Delta$ for all $j = 1, \ldots, M$. The variances initial values is concurrence with the fact that in the initial stage of the algorithm it is better for the variance to be too large rather than too small. This is to ensure that each line component is responsible for a reasonable fraction of the feature points. Otherwise, if a line component is not responsible to any feature point its parameters will not changed by the EM algorithm.

[2] Note that we can not supply an analytical proof for this statement.

[3] The "hat" is omitted to simplify notation.

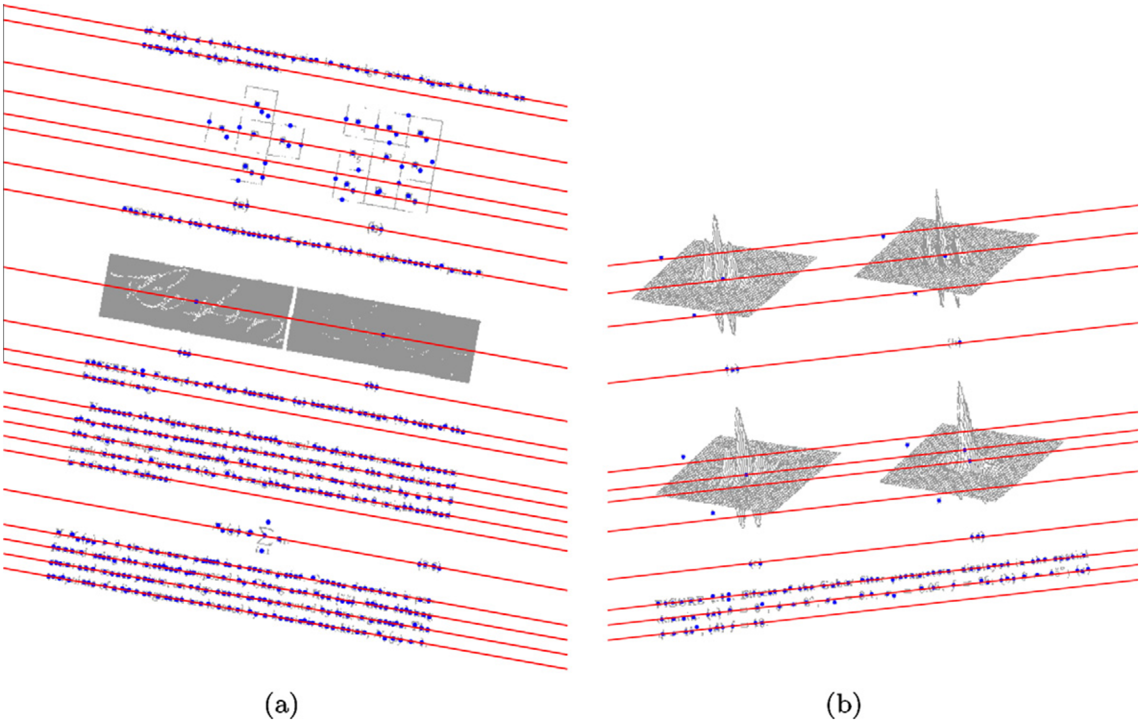


Fig. 4. Skew results of rotated images contain printed text with graphics. Each subfigure shows the centroids of the connected components which used as feature points and the estimated model lines. (a) Image rotated at −10°, the estimated skew angle was −10.17°. The ground-truth skew angle of the non-rotated image was $\phi_{gt} = -0.241^\circ$, hence, the absolute error for this rotated image was $|-10.17-(-10+\phi_{gt})| = 0.07^\circ$; (b) image rotated at 7°; the estimated skew angle was 6.5°. The ground-truth skew angle of the non-rotated image was $\phi_{gt} = 0.32^\circ$, hence, the absolute error for this rotated image was $|6.5-(7+\phi_{gt})| = 0.82^\circ$. The large error is cause by the lack of text lines in the image.

3.5. Complexity

In each iteration, of the EM algorithm, the posterior distribution α_{jn} is to be calculated for each line according to Eq. (10). This calculation requires $\mathcal{O}(N_{EM}M^2)$ where N_{EM} is the number of feature points and M is the number of components (lines) in the mixture model. The other mixture parameters $\{\pi_j\}_{j=1}^M, \left\{\sigma_j^2\right\}_{j=1}^M, \{A_j\}_{j=1}^M$ and $\{B_j\}_{j=1}^M$ require $\mathcal{O}(N_{EM})$ operations each (Eqs. (11), (14) and (13)) for a total of $\mathcal{O}(N_{EM}M)$.

The projection profiles method required for each projection $\mathcal{O}(N_P)$ operations (note that usually $N_P \gg N_{EM}$. For example in one of our test images $N_P \approx 54000$ and $N_{EM} \approx 680$. Then a rotation procedure or other calculation need to be done to calculate the projection in other angles. The final complexity term is duo to the criterion function calculation which can be very simple as the sum of squares or more complex involving fourier coefficient and other features.

The experiments were performed on a 64-bits Linux machine with 1.73 GHz Intel Dual Core CPU (T2370) and 2 GB of memory. We implemented both algorithms in Matlab without too much optimization effort. The average running time for an image with about 1600 feature points and a model with 30 lines was around 5 s, which is similar to the running times of the projection profiles implementation. As discussed in the next section, downsampling was applied to the image for the projection profiles implementation in order to reduce the running time, to be comparable to the EM implementation.

4. Experimental results

4.1. Synthetic data

A set of 200 points was synthesized from the distribution

$$p(y_n) = 0.5\mathcal{N}(10+2x_n,\sigma_1^2) + 0.5\mathcal{N}(80-x_n,\sigma_2^2).$$

where $\sigma_1 = 20$ and $\sigma_2 = 15$. We initiated the parameters at $a_1^{(0)} = 56.6, b_1^{(0)} = 0, a_2^{(0)} = 132.3$ and $b_2^{(0)} = 0$, which correspond to two parallel horizontal lines. The variances were initiated at $\sigma_1^{(0)} = \sigma_2^{(0)} = 75.6$. This choice is a wide variance according to the same considerations described above (Section 3.4).

The estimated parameters were $\hat{a}_1 = 13.75,\ \hat{b}_1 = \ 1.9,\ \hat{\sigma}_1 = 17.3,\ \hat{a}_2 = 77.6,\ \hat{b}_2 = -0.91,\ \hat{\sigma}_2 = 16.7$ and $\hat{\pi} = 0.52$, after 17 EM iterations. Fig. 1 shows the scatter of the data points together with the estimated lines and the true lines.

4.2. Text document skew results

To test our method, we used two data sets. The first dataset consists of 50 binary images scanned at 300 dpi. The images from this data set were taken from a number of scientific text books and journals and were binarized using the well known Otsu method (Otsu, 1979). They contain both pure text information of various fonts/sizes and mixture of text, graphics, tables, mathematical expressions and pictures. Example of two rotated images from this data set are shown in Fig. 4, as well as the estimated model lines. These results were obtained using the parallel lines model (16) with constant variance (15). The other data set consists of 50 binary images of Hebrew calligraphic manuscripts extracted from our database (Bar-Yosef et al., 2007). These historical document images were binarized using the multi-stage thresholding algorithm presented in (Bar-Yosef, 2005). The images from this data set contain handwritten textual components with a small amount of noise components due to binarization faults. Example of two rotated images from this data set are shown in Fig. 5, as well as the estimated model lines and the line slopes distribution. These results were obtained using the model with the no-intersection constraint.

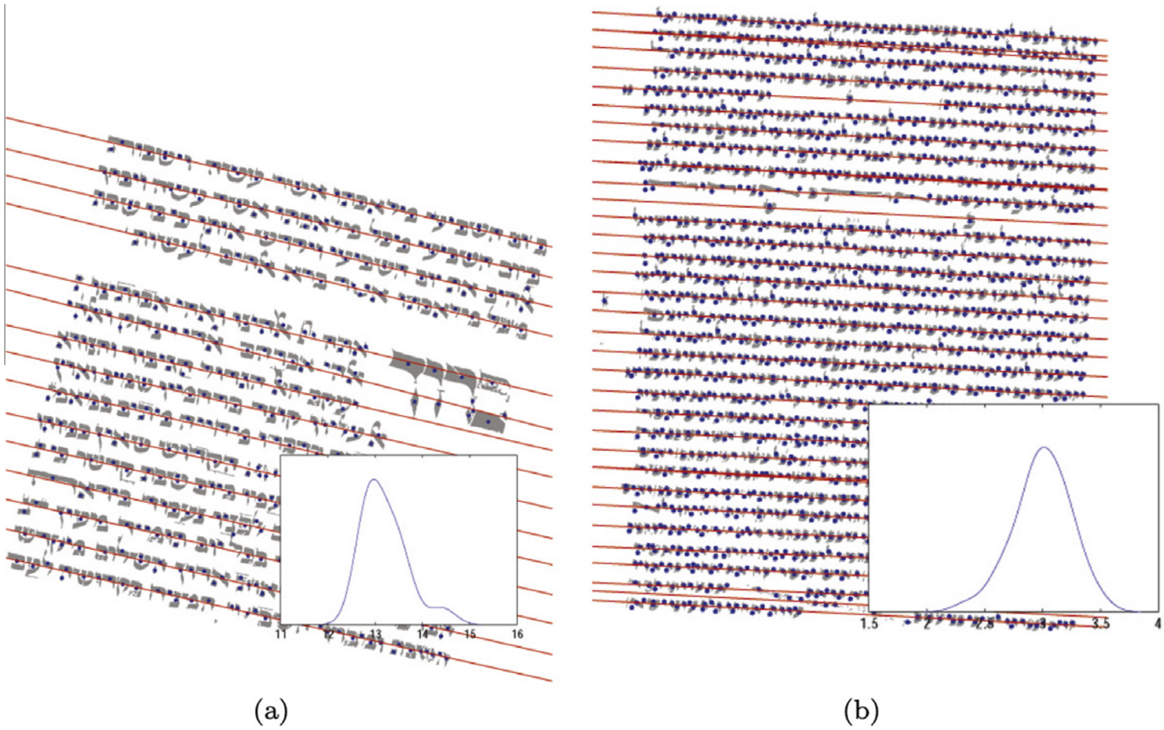


Fig. 5. Skew detection results for rotated images of Hebrew calligraphic handwriting (STAM) documents. The results were obtained using the no-intersections constraint. Each subfigure shows the centroids of the connected components which used as feature points, the estimated model lines and the slope angles distribution. (a) Image rotated at $-13°$, the detected skew angle $-12.97°$; (b) image rotated at $-3°$, the detected skew angle $-3.01°$. The ground-truth skew angle of the non-rotated images was 0° for both images. Hence, the absolute error for (a) was 0.03° and for (b) was 0.01°.

All test images had only one text column. Document images with more than one column need a preprocessing stage to separate the text columns. This can be obtain easily since column of text are usually well separated. In this evaluation experiment we assumed that the images are separated into columns.

We compare our method to a projection profiles based algorithm, similar to the method proposed by Bloomberg et al. (1995). We choose this method because projection profiles is a general and popular tool for skew detection and numerous algorithm are based on it. We implemented a downsampling scheme that preserved horizontal structure in the image similar to Bloomberg et al. (1995). The downsampling begins by tiling the image into $N \times N$ pixel cells. In each tile, one of the N rows of pixels is chosen arbitrary. If any pixel in that row is '1', the output pixel is '1'; otherwise it is '0'. The downsampling factors N were adjusted to match the running times of the competitive algorithms for each data set. For the first data set $N = 8$ and for the second $N = 16$.[4] The difference is due to the different ratio of extracted feature points to foreground pixels in the images. The skew angle estimate was obtained using a course-to-fine search algorithm. The algorithm first calculates projection profiles over a sequence of angles that have a course resolution. The angle that maximizes a criterion function[5] is used as the center for a finer resolution search for the skew angle. This recursive subdivision is repeated until the final decision is reached. By this procedure, a final resolution of 0.01° is achieved.

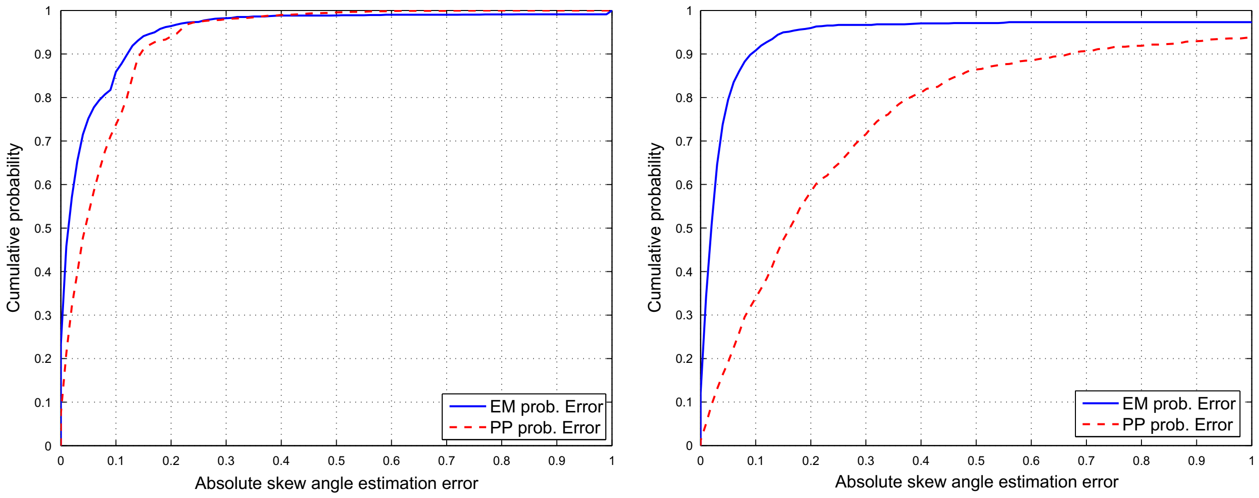
For the EM algorithm we used the centroids of the connected components as feature points. The original images were used since no improvement in computational time had been achieved by downsampling. For this evaluation experiment, we implemented a two stages optimization process. The initial state for the first phase in all the reported experiments were horizontal parallel lines as been described in Section 3.4. In the first stage, the EM algorithm was applied using the update Eqs. (11), (13) and (14) and using the no-intersection constraint. The first stage was run for a maximum of 30 iterations. The aim of the first stage is to get a course estimate of the skew angle and to bring the model parameters near the local minima of the desired optimum. The second stage used the first stage estimated model as initial state and run until convergence. For this stage we used the parallel lines model (16) with constant variance (15) for all lines. The final skew angle was defined as the angle of the lines slope.

Every image was rotated through every angle from $-10°$ to $10°$ in 0.5-degree steps. This becomes a total population of $2050 = 50 \times 41$ test images for each of the two classes of document images. The tested range was chosen to be -10 to 10 since this is a practical requirement from a skew detection system since larger skew angles are not frequent. We assumed that the anticipated number of model lines is given either by automatic selection (Section 3.3) or manually. Note that this number is usually more than the actual text lines in the image.

For each test image we compared the output of the two algorithms, the proposed algorithm and the projection profiles, to a ground-truth angle. The ground-truth angle was extracted from the original (non-rotated) image. It been defined to be the output of our algorithm with parallel lines model (16) and with the optimal number of model lines manually selected. The ground-truth angles for the rotated images were the angle of the non-rotated image plus the artificial rotation angle (see Figs. 4 and 5). We chose this angle as ground-truth for several reasons. First, as been noted in (Chen and Haralick, 1994), manually setting the ground-truth skew angle is actually an estimate which is subject to error and thus biases the results. Moreover, the projection profiles method is not adequate for small angles, it cannot identifies small skew angles (Bloomberg et al., 1995). For this reason, the skew angle in (Bloomberg et al., 1995) was obtained according to the output of

[4] Even with higher resolution our algorithm outperformed the projection profiles method on the handwritten documents data set.

[5] See Bloomberg et al. (1995) for a discussion on the choice of the criterion function.



(a)

(b)

Fig. 6. The probability distributions of the absolute skew estimation errors for our EM based algorithm vs the projection profiles algorithm. (a) The results for the printed document images data set; (b) the results for the STAM handwriting document images data set.

a few rotated images. Finally, the algorithm output with the described configurations yielded accurate results for all tested images.

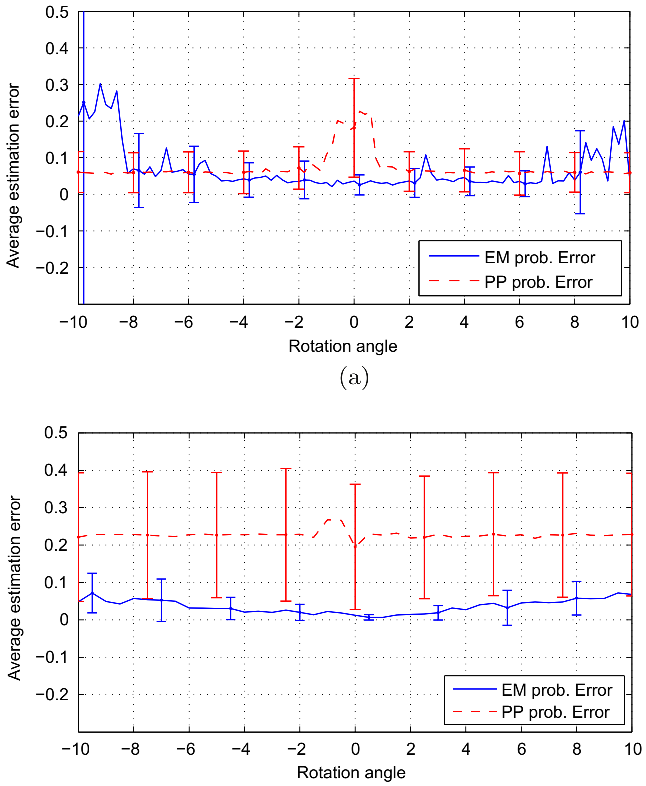
For each data set we measured the probability distribution of the skew angle estimation, i.e. the difference between the ground-truth and the estimated skew angle, obtained by the two algorithms. Let $\hat{\phi}$ denote the text skew angle detected by one of the algorithms and let $\hat{\phi}_T$ denote the ground-truth skew angle. The cumulative probability distribution of the absolute skew estimation errors is computed by $Prob(|\hat{\phi} - \hat{\phi}_T| \leqslant x)$. The probability distributions of the skew angle estimation error are plotted in Fig. 6 for both data sets and for the both algorithms.

The experimental results demonstrated the advantages of our method to skew estimation of document images. On the first data set (printed document images) our algorithm detected skew angles which were within 0.1° of the ground-truth skew angles at a probability of about 86% while the projection profiles probability for the same accuracy dropped to about 74%. On the other data set (STAM handwriting images) our algorithm exhibits very good performance compared to the projection profiles method. The probabilities that the estimated skew angles lie within 0.1° of the ground-truth skew angle were 90% and 34% for our algorithm and for the projection profiles respectively.

There are several reasons for the fact that our algorithm outperforms the projection profiles method for the STAM data set. First, the STAM text lines are more dense than printed text lines which affect negatively on the analysis of the projection profiles. Second, the criterion function that we used might be suited for printed text only and/or to Latin script only. Finally, handwritten documents possess more variability than printed ones. This feature is the foundation of our method and on the other hand, deteriorates the projection profiles method results.

There is another disadvantage of the projection profile algorithm compared to the proposed algorithm. In the projection profile method the skew angle is searched within a discrete set of angles and, the algorithm needs to calculate the projection profile for each angle in that set. In the proposed algorithm there is no set of predefined angles and the number of iterations depend on the data and the skew angle.

Resolution reduction is common in skew detection methods that are based on projection profiles, although it increases the error rates, due to the computational cost of this method. In our method, image resolution reduction is not required and the complexity is proportional to the selected feature points. However, resolution reduction can help in the connected components labeling and feature extraction stage.



(a)

(b)

Fig. 7. The average absolute estimation error for each rotation angle. The solid line is the result of our EM-based algorithm whereas the dash line is the result of the projection profiles algorithm. Error bars indicate the standard deviation of the estimation error. (a) Results for the printed text document images; and (b) for the calligraphic data set.

Fig. 7 presents the average absolute estimation error to rotation angle for both methods and both data sets. The error-bars indicate one standard deviation (std) from the mean. In each pair of error-bars, the right correspond to the EM algorithm and the left to the

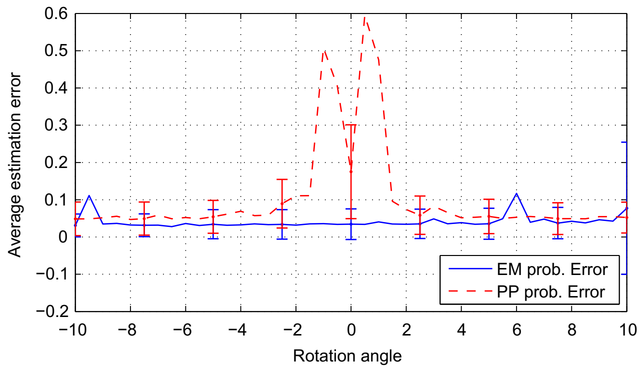


Fig. 8. The average absolute estimation error for each rotation angle. The solid line is the result of our EM-based algorithm whereas the dash line is the result of the projection profiles algorithm. Error bars indicate the standard deviation of the estimation error. This figure presents the results based on a simple model selection approach, where the parameters of several model with different number of lines were estimated and the model with the lowest likelihood (2) was selected.

projection profiles method. It can be noticed that for the printed text data set, the projection profiles method is inaccurate for small angles, i.e. -2 to 2. However, for larger skew angles its estimation is more consistent than the EM (lower std). On the other hand, for the calligraphic text images the EM algorithm outperformed the projection profiles method for any rotation angle.

Although our method is not as accurate as the projection profiles for skew angles larger than ±8 (for the printed text data), it provides a confidence measure for the skew estimation. The confidence measure is based on the log likelihood of the model (2). To demonstrate this idea we returned to the above experiment on a similar dataset consisting of binary images of printed text. This time, for each image we estimate n mixture models with increasing number of lines, and the estimated skew angle was extracted from the model with the minimum log likelihood. For this experiment we choose $n = 5$. The results are reported in Fig. 8. The improvement compared with Fig. 7(a) is evident.

5. Discussion and conclusions

We proposed an EM-based algorithm for skew angle detection of text images. We used the EM algorithm to estimate the parameters of a mixture model where each component is a Gaussian distribution of the vertical offsets from a straight line. Usually for one line this is solved using the linear least squares method. However, for multiple lines the least squares method is not applicable since the feature points need to be segmented into line representative groups. The EM algorithm can iteratively solve this problem and yield an estimate of the parameters of multiple lines model. We presented a few adjustments of this general mixture model in order to suit the skew detection problem. The modifications include, a mixture parallel line model that all components have the same variance or a general mixture of line model that is constrained not to have intersecting lines. Finally, The skew angle estimate is obtained from the slope angle histogram of the detected lines. We applied the proposed algorithm to scanned printed documents and to Hebrew calligraphic handwritten documents with satisfactory results. We found that the variability of the handwriting character features can be modeled as a straight line in Gaussian noise.

We would like to address three issues regarding the implementation of the proposed algorithm. First, similar to any other linear/Gaussian model the problem of estimation in the present of outliers should be addressed. This problem is not critical if we only after the global skew angle. If we have a reasonable number of text lines in the image, using kernel density estimation the results can give a good estimate even if a few lines are attracted by outliers. This is demonstrated in Fig. 3(b) where a few estimated slopes deviate from the main mode of the estimated distribution and do not influence the final skew angle estimate.

The second issue regards the use of an iterative procedure for finding the maximum of the likelihood function. This is the known problem of being trapped in a local maximum and is common in almost any optimization method. However, using the no-intersections constraint obligates the EM algorithm to converge to a local maximum that better suits the skew estimation problem.

Finally, the number of components or lines that have to be estimated in the mixture model should be addressed. This is a major drawback when using a mixture model. In order to solve this problem we introduced a special criterion for choosing the optimal number of lines in the mixture model which is based on the estimation variance. In addition, numerous experiments showed a strong correlation between low estimation variance and skew angle estimation accuracy. This is a unique feature of our algorithm. It provides a confidence measure on the estimation accuracy, which can be used to correct the model configuration in order to obtain the best estimation.

We compared our method to a projection profiles based algorithm. Our algorithm achieved better error rates for printed document images as well as for Hebrew calligraphic handwritten document images. For the handwritten documents, our algorithm outperformed the projection profiles. The probabilities that the estimated skew angles were within 0.1° of the ground-truth skew angle were 90% and 34% for our algorithm and for the projection profiles respectively. Moreover, in comparison to the horizontal profiles methods our method has several additional advantages. First, we do not need to search over a discrete set of skew angles. Second, the proposed method is more computationally efficient than the projection profile method which required to rotate all feature points for all the considered skew angle candidates and thus required a resolution reduction. In addition, by a selective feature selection our algorithm can detect line features such as bottom/up of text lines.

Finally, The proposed method is very flexible, in that we can use different feature points for different writing styles or languages, e.g. centroids of connected components, top of connected components and so on.

Acknowledgments

This work was partially supported by the Paul Ivanier Center for Robotics and Production Management at Ben-Gurion University, Israel, and in part by ISF Grant 1266/09, “Algorithms for historical document analysis in Hebrew and Arabic”.

References

Akaike, H., 1974. A new look at the statistical model identification. IEEE Trans. Automatic Control 19 (6), 716–723.

Amin, A., Fischer, S., 2000. A document skew detection method using the hough transform. Pattern Anal. Appl. 3 (3), 243–253.

Bagdanov, A., Kanai, J., 1996. Evaluation of document image skew estimation techniques. In: Vincent, L.M., Hull, J.J. (Eds.), Proc. SPIE, Document Recognition III, vol. 2660, pp. 343–353.

Baird, H.S., 1987. The skew angle of printed documents. In: Proc. 1987 Conf. of the Society of Photographic Scientists and Engineers, pp. 21–24.

Bar-Yosef, I., 2005. Input sensitive thresholding for ancient hebrew manuscript. Pattern Recognition Lett. 26, 1168–1173.

Bar-Yosef, I., Beckman, I., Kedem, K., Dinstein, I., 2007. Binarization, character extraction, and writer identification of historical hebrew calligraphy documents. Int. J. Doc. Anal. Recognition 9 (2–4), 89–99.

Bar-Yosef, I., Hagbi, N., Kedem, K., Dinstein, I., 2008. Fast and accurate skew estimation based on distance transform. In: The Eighth International Workshop on Document Analysis Systems, pp. 402–407.

Bishop, C.M., 1995. Neural Networks for Pattern Recognition. Clarendon Press, Oxford.

Bloomberg, D.S., Kopec, G.E., Dasari, L., 1995. Measuring document image skew and orientation. In: Vincent, L.M., Baird, H.S. (Eds.), Proc. SPIE, Document Recognition II, vol. 2422, pp. 302–316.

Bouchard, G., Celeux, G., 2006. Selection of generative models in classification. IEEE Trans. Pattern Anal. Machine Intell. 28 (4), 544–554.

Cao, Y., Wang, S., Li, H., 2003. Skew detection and correction in document images based on straight-line fitting. Pattern Recognition Lett. 24 (12), 1871–1879.

Chen, S., Haralick, R.M., 1994. An automatic algorithm for text skew estimation in document images using recursive morphological transforms. In: ICIP (1), pp. 139–143.

Dempster, A., Laird, N., Rubin, D., 1977. Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Statist. Soc. 39 (Series B), 1–38.

Duda, R.O., Hart, P.E., 1972. Use of the hough transforms to detect lines and curves in pictures. In: Proc. ACM. published as Proc. ACM, 15(1), pp. 11–15.

Egozi, A., Dinstein, I., 2007. An EM based algorithm for skew detection. In: Ninth Internat. Conf. on Document Analysis and Recognition (ICDAR), vol. 1, pp. 277–281.

Hashizume, A., Yeh, P.S., Rosenfeld, A., 1986. A method of detecting the orientation of aligned components. Pattern Recognition Lett. 4, 125–132.

Hull, J., 1998. Document image skew detection: Survey and annotated bibliography. In: Hull, J., Taylor, S. (Eds.), Document Analysis Systems II. World Scientific, pp. 40–64.

Likforman-Sulem, L., Zahour, A., Taconet, B., 2007. Text line segmentation of historical documents: a survey. Internat. J. Doc. Anal. Recognition 9 (2), 123–138.

Liolios, N., Fakotakis, N., Kokkinakis, G., 2001. Improved document skew detection based on text line connected-component clustering. Internat. Conf. Image Process. 1, 1098–1101.

Lu, S., Wang, J., Tan, C., 2007. Fast and accurate detection of document skew and orientation. Internat. Conf. Doc. Anal. Recognition 2, 684–688.

Otsu, N., 1979. A threshold selection method from gray level histograms. IEEE Trans. Systems Man Cybernet. 9, 62–66.

Postl, W., 1986. Detection of linear oblique structures and skew scan in digitized documents. Internat. Conf. Pattern Recognition, 687–689.

Schwarz, G., 1978. Estimating the dimension of a model. The Ann. Statist. 6, 461–464.

Sheather, S.J., Jones, M.C., 1991. A reliable data-based bandwidth selection method for kernel density estimation. J. Roy. Statist. Soc. Ser. B (Methodol.) 53 (3), 683–690.

Smith, R., 1995. A simple and efficient skew detection algorithm via text row accumulation. In: Proc. Third Internat. Conf. on Document Analysis and Recognition, vol. 2, pp. 1145–1148.

Srihari, S.N., Govindaraju, V., 1989. Analysis of textual images using the hough transform. Machine Vision Appl. 2, 141–153.

Yu, C.L., Tang, Y.Y., Suen, C.Y., 1995. Document skew detection based on the fractal and least squares method. In: Proc. Third Internat. Conf. on Document Analysis and Recognition, pp. 1149–1152.