Pattern Recognition 43 (2010) 937–948

Contents lists available at ScienceDirect

Pattern Recognition

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

# e-PCP: A robust skew detection method for scanned document images

Prasenjit Dey, S. Noushath*

*Hewlett-Packard Laboratories, #24 Salarpuria Arena, Koramangala, Bangalore 560030, India*

ARTICLE INFO

*Article history:*
Received 20 April 2009
Accepted 24 June 2009

*Keywords:*
Skew detection
Piece-wise coverings (PCP)
Confidence measure
Text-flow orientation detection
Enhanced-PCP
Skew correction

ABSTRACT

We present here an enhanced algorithm (e-PCP) for skew detection in scanned documents, based on the work on Piecewise Covering by Parallelogram (PCP) for robust determination of skew angles [C.-H. Chou, S.-Y. Chu, F. Chang, Estimation of skew angles for scanned documents based on piecewise covering by parallelograms, Pattern Recognition 40 (2007) 443–455]. Our algorithm achieves even better robustness for detection of skew angle than the original PCP algorithm. We have shown accurate determination of skew angles in document images where the original PCP algorithm fails. Further, the increased robustness of performance is achieved with reduced number of computation compared to the originally proposed PCP algorithm. The e-PCP algorithm also outputs a confidence measure which is important in automated systems to filter cases where the estimated skew angle may not be very accurate and thus can be handled by manual intervention. The proposed algorithm was tested extensively on all categories of real time documents and comparisons with PCP method is also provided. Useful details regarding faster execution of the proposed algorithm is provided in Appendix.



## 1. Introduction

Skew estimation of document refers to the process of finding the angle of inclination made by the document with respect to horizontal axis, which is often introduced during document scanning. For any ensuing document image processing tasks (such as page layout analysis, OCR, document retrieval etc.) to yield accurate results, the skew angle must be detected and corrected beforehand. Though a large number of skew estimation methods have been proposed, development of a solitary skew estimation algorithm which is relatively fast and yet handles wide range of documents is still an elusive goal. It is because of this very reason, document skew estimation research is still active although it has been studied for several decades now.

The algorithms for skew estimation can mainly be classified as the ones based on (i) projection profile (PP) [6,3], nearest neighbor (NN) [1,10], (iii) Hough transform (HT) [9,4,7,12] and (iv) cross-correlation (CC) [8,5].

Among these methods, PP based methods are most commonly used. These methods compute projection profiles of the document at various angles and compute the skew angle of the document based on some maximization criteria. However, these methods are computationally intensive and they need to carry out expensive rotation operation at every angle. Moreover, it is sensitive to the layout of the image [6]. The NN techniques calculate skew angle between each component and its nearest neighbor and the histogram of the angles are formed thereby. The peak in the histogram corresponds to the skew angle of the document. Another class of skew estimation methods is based on the HT. The idea is that collinear pixels in Cartesian space constitutes cluster of $(\rho, \theta)$ bins in Hough space. The peak in the Hough space corresponds to skew angle of the document. However, there are two main demerits of both NN and HT based methods:

(1) one has to extract text regions from the document which is again a non-trivial task for documents whose layouts are complex and
(2) they are computationally intensive.

On the other hand, there are also methods that compute skew angle of the document based on maximum variance of transition counts (TC) [11] and based on cross-correlations (CC).

Recently a robust skew estimation algorithm based on piecewise coverings of objects by parallelograms (PCP) was proposed [2]. In this approach, the document image is divided into several non-overlapping slabs and the objects within each slab is covered by parallelograms at various angles. The angle at which objects are *best covered* corresponds to skew angle of the document. The PCP algorithm has been demonstrated to achieve faster and robust results than PP, HT and NN based methods in [2].

However, there exists one main drawback with this approach. When vertically flowing text (VFT) in a document (which is common in Chinese and Japanese documents) touches the borders of

* Corresponding author. Tel.: +91 80 25042256.
*E-mail addresses:* pdey@hp.com (P. Dey), nawali_naushad@yahoo.co.in, noushath.s@hp.com (S. Noushath).

0031-3203/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2009.06.019


**Fig. 1.** A document with vertical flowing text touching the borders.


**Fig. 2.** Parallel scan lines drawn at an angle of text line skew.

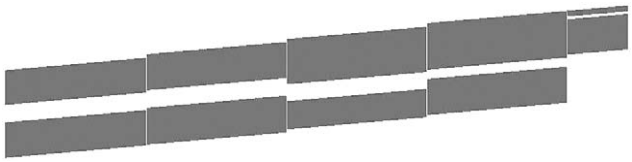

**Fig. 3.** Parallelograms constructed for the text present in Fig. 2.

the document as shown in Fig. 1, this method fails to yield desired skew angle. This is because the maximization criterion (this will be discussed in Section 2) of PCP approach [2] will not *strongly* favor a particular angle to arrive at the actual estimate of the skew. Consequently, the method may lead to a wrong estimate of the skew angle for such kinds of documents.

Moreover, one can encounter documents of type shown in Fig. 1 very frequently, especially while scanning large documents (such as newspapers, posters etc.) whose content often goes beyond the scanner bed. This necessitates the need of a mechanism in the original PCP approach, which automatically determines the flow of text and then determines the slab orientation accordingly (i.e. either horizontal or vertical). Hence, our proposed enhanced-PCP (e-PCP) algorithm enhances the conventional PCP in this aspect. In contrast to the PCP method [2], the overall enhancements achieved in the proposed method are as follows:

(1) Improved robustness across any kind of documents, especially for VFT documents.
(2) Reduction in number of computations and yet retaining the accuracy of the algorithm.
(3) Insensitive to size of the slab widths by automatically determining the slab orientation.
(4) A robust confidence measure module for reliable skew estimation, which is useful in automated document processes.

Rest of the paper is organized as follows: Review of the PCP algorithm and its shortcomings are given in Section 2. The proposed e-PCP algorithm and its computational load are described in Section 3 and Section 4 respectively. Experimental results are presented in Section 5, and we finally draw some conclusions in Section 6.

## 2. Review of the PCP algorithm

In this section we briefly review the PCP algorithm [2]. This algorithm is based on the concept that document contains many rectangular objects (text lines, text regions, forms, rectangular pictures etc.) and when there is no skew in the document, these objects can be *best covered* by rectangles. On the other hand, when there is a skew in the document, these objects can only be *best covered* by parallelograms.

In the process, the document is first divided into a number of non-overlapping vertical slabs, and scan-lines are drawn at all angles within the skew angle range (e.g. $-15^\circ$ to $+15^\circ$). Each scan line is further divided into as many sections as the number of slabs, where a section refers to a part of the scan line within a slab. Fig. 2 demonstrate the process of dividing images into slabs and drawing scan lines.[1] In this example, since width of the image is not a multiple of slab width, the last slab is small compared to others. Each section of the scan line is examined for occurrence of any black pixel. If a section contains at least one black pixel, all pixels along that section will be changed to gray, else it will be counted as a white section. Fig. 3 shows parallelograms constructed for the objects shown in Fig. 2 by changing those sections to gray which contain at least one black pixel.

This process of scan-line drawing is repeated for all angles within the skew angle range and number of white sections at each angle is computed. The intuitive idea is that when scan lines are drawn at angle corresponding to skew angle of the document, there will be more number of white sections than the gray ones, which is quite evident in Fig. 4. This is true even in case of complex cases such as when document has large scale figures, forms or tables, multilingual documents, etc. [2]. Thus the process of estimating skew angle reduces to maximizing the following criteria:

$$\theta^* = \underset{\theta}{\operatorname{argmax}}\, WS(\theta) \tag{1}$$

where $WS(\theta)$ is the number of white sections when scan lines are drawn at angle $\theta$.

Unlike HT, PP or NN based algorithms, this method produces robust results for many real time documents [2]. Nevertheless, it suffers from following two major drawbacks which deserves further study:

(1) *Subjectiveness of slab width*: As mentioned earlier, when text lines are aligned vertically and if their content touches borders of the document, the success of the algorithm depends on the appropriate size of the slab width. For example, the document shown in Fig. 5 has a skew angle of $0^\circ$. The estimated skew angle for the document for different slab widths is shown in Table 1. We see that the estimated skew angle is highly dependent on the slab width, which is fixed to a particualr value a priori in PCP method [2]. Hence, for such kind of documents, determining appropriate slab width is highly subjective in nature.
(2) *Number of computations*: The whole process of computing the number of white section has to be repeated for each skew angle in the given skew range. In future, if the skew range has to

[1] For illustration purpose, scan lines are drawn at an angle corresponding to the skew angle of the text, and with some gap between them.

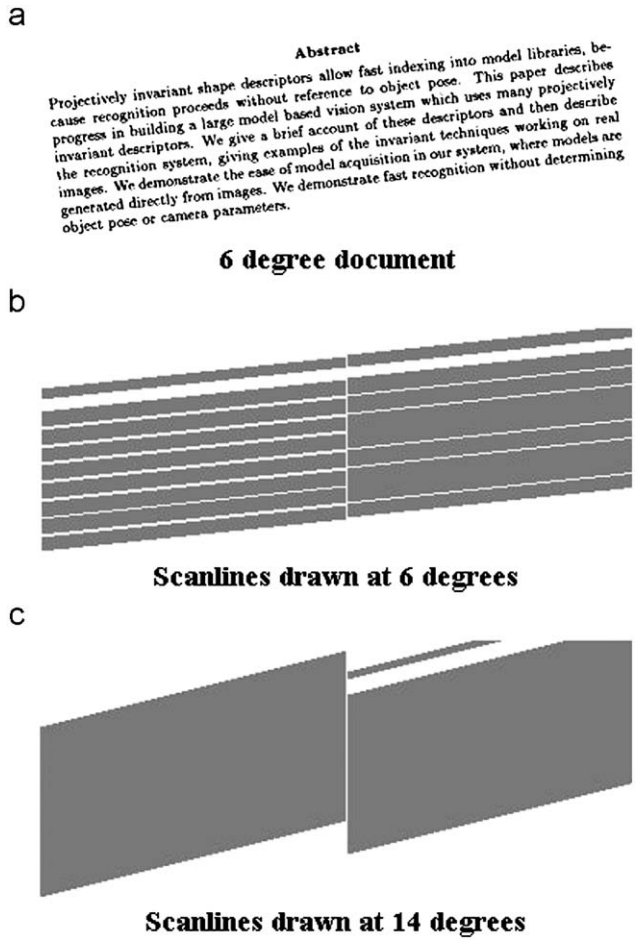


**Fig. 4.** Parallelograms constructed for a 6° document at two different angles.


**Fig. 5.** Illustrating subjectiveness of slab width and a failure case of PCP algorithm because of slab width.

**Table 1**
Table of estimated skew angle for Fig. 5 with different slab widths.

| Slab width | Estimated angle (deg) |
|---|---|
| 10 | 3.0 |
| 20 | 1.5 |
| 50 | 0.5 |
| 100 | 0.19 |
| 150 | 0.09 |
| 200 | 13.5 |

be increased, the computational burden increases significantly. Although in [2] some optimal search process has been suggested to reduce the number of computations, there is still some scope to improve the computational efficiency by retaining the same robustness of the algorithm.

In this paper, the proposed e-PCP method not only resolves the aforementioned drawbacks of PCP but also provides a good confidence measure useful for real time automated skew correction mechanisms.

## 3. Proposed e-PCP algorithm

The proposed e-PCP algorithm can robustly estimate the skew angle for both kinds of documents: (a) horizontal-flowing text (HFT) document (such as a typical English document) and (b) vertical-flowing text (VFT) document (such as a Chinese/Japanese document as shown in Fig. 1). It is particularly interesting in cases, as shown in Fig. 5, where the original PCP algorithm completely fails to determine the skew angle whereas the e-PCP performs quite well. This robustness of skew angle estimation is achieved with reduced number of computations in e-PCP compared to the PCP algorithm [2].

### 3.1. Overview

The main cause of failure of PCP algorithm on cases like the one shown in Fig. 5 is due to the lack of information about the flow of text in the document. A HFT document with vertical slabs and its corresponding scan lines, result in areas of white spaces which change rapidly from being maximum when the scan lines are completely aligned to text flow, to being minimum when the scan lines are mis-aligned to the text flow. This is as illustrated in Fig. 6(a). However, if we consider a VFT document with the same slab orientation (i.e., vertical slab) with corresponding scan lines as shown in Fig. 6(b), there is very little variation of the area of white sections for reasonable variation in angles of the scan lines.

Let $\{WS(\theta)\}$ be the set of computed white sections for different angles in a particular slab orientation. We compute the variance of these $\{WS(\theta)\}$ for the particular slab orientation. Table 2 shows the variance of the computed areas of the white sections for both vertical and horizontal slab orientations. We observe that the variance of the area of white sections is always higher for slab orientation that is actually *correct* in the sense that the sensitivity of white section areas is high for change in scan-line angles $\theta$ for that document.

The *correct* orientation of the slab is the use of *vertical slab* on an HFT document and *horizontal slab* on a VFT document.

Thus we compute a coarse variance of white section area of the image for different slab orientation and determine the correct slab orientation. The correct slab orientation is used for further analysis and estimation of the skew angle. Hence in Fig. 5 the cause of failure to estimate the skew angle is because of the situation in Fig. 6(b). The white section area variation is almost insensitive to the skew angle variation for vertical slab orientation. Hence it is very difficult

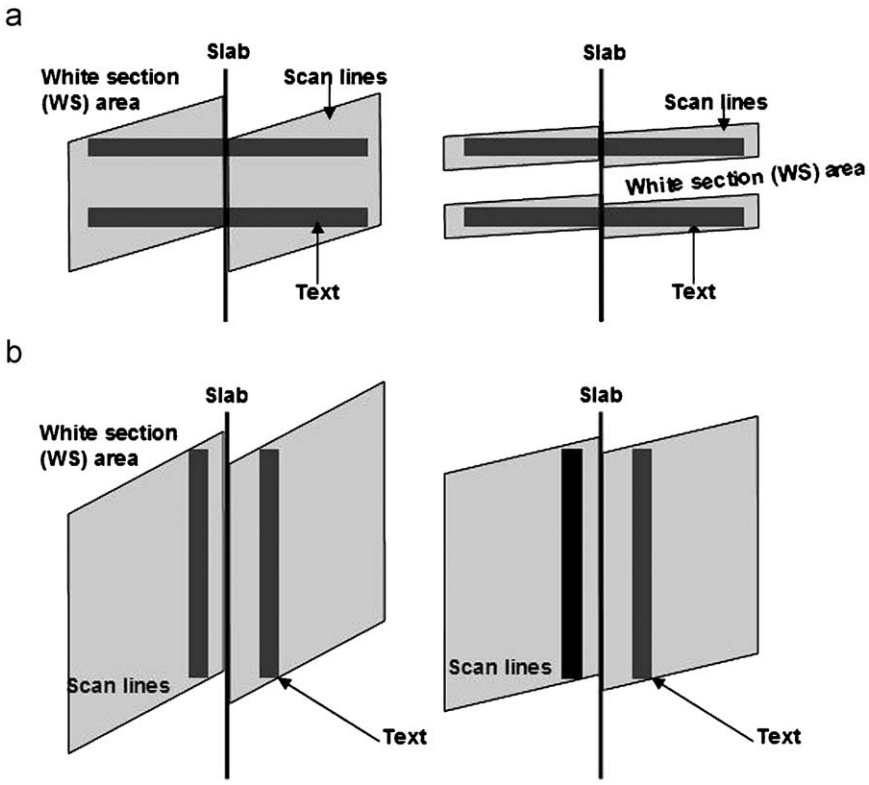


**Fig. 6.** White section area for different text flows and slab orientations.

**Table 2**
A table of variance values of white section area for each of five HFT and VFT document images.

| Document | Variance (for vertical slabs) | Variance (for horizontal slabs) |
|---|---|---|
| 1 (HFT) | 141.58 | 91.29 |
| 2 (HFT) | 135.72 | 120.91 |
| 3 (HFT) | 115.76 | 105.18 |
| 4 (HFT) | 153.88 | 87.77 |
| 5 (HFT) | 164.37 | 79.57 |
| 6 (VFT) | 192.48 | 453.57 |
| 7 (VFT) | 67.55 | 206.23 |
| 8 (VFT) | 27.61 | 34.75 |
| 9 (VFT) | 38.79 | 49.94 |
| 10 (VFT) | 126.29 | 367.22 |

to distinguish a distinctively large white section area for a particular scan-line angle compared to the other scan-line angles. However, if a horizontal slab is applied to the document it has a large variance in white section area for each scan-line angle as in Fig. 6(a) and thus can be used to distinguish a clearly large white section area for a particular scan-line angle.

### 3.2. The algorithm

The flow diagram of the complete skew estimation process is shown in Fig. 7. Given an input skewed document, it passes through preprocessing steps such as down-sampling, binarization and edge detection. Down-sampling is used to speed up the skew estimation process in case of large documents. The edge detection step is an optional step and we have retained it in our system in order to handle input documents with arbitrary pixel values, foreground being dark and background being light or vice versa. The important steps involved in the e-PCP algorithm are:

(1) Coarse level estimate of skew angle and text flow determination.
(2) Confidence measure.
(3) Finer level estimate of skew angle.

The aforementioned steps of the algorithm are detailed in subsequent sections.

#### 3.2.1. Coarse level estimate of skew angle and text flow determination

Suppose that the range of skew estimation is $[-15^\circ \ +15^\circ]$ (this range of values have been taken for our experiments but the proposed method is general enough to apply for any skew ranges). This step computes the number of white sections obtained by drawing scan lines at every angle from $-15^\circ$ to $+15^\circ$ in steps of $5^\circ$. This essentially obtains seven values of white sections across the skew range $[-15^\circ \ +15^\circ]$ (i.e. for every angle at $-15^\circ$, $-10^\circ$, $-5^\circ$, $0^\circ$, $5^\circ$, $10^\circ$, $15^\circ$). Compute the white section areas for the range of angles given above for a vertical slab orientation. The variance is computed for the seven values of the white section area. Let the variance of these values be $Var_1$.

We again compute the white section areas for the range of angles given above for a horizontal slab orientation. The variance is computed for the seven values of white section area corresponding to the angles. Let the variance of these seven values be $Var_2$. The document is classified as an HFT document if $Var_1$ is greater than $Var_2$, else it is classified as a VFT document. Let $\theta_{coarse}$ be the scan-line angle that yielded maximum white sections, which is the coarse level estimate of skew. Once the direction of text flow is determined, for robust results, the subsequent steps of computing white sections are applied according to the computed slab orientation (horizontal slab or vertical slab). In contrast to the existing PCP method [2], the coarse

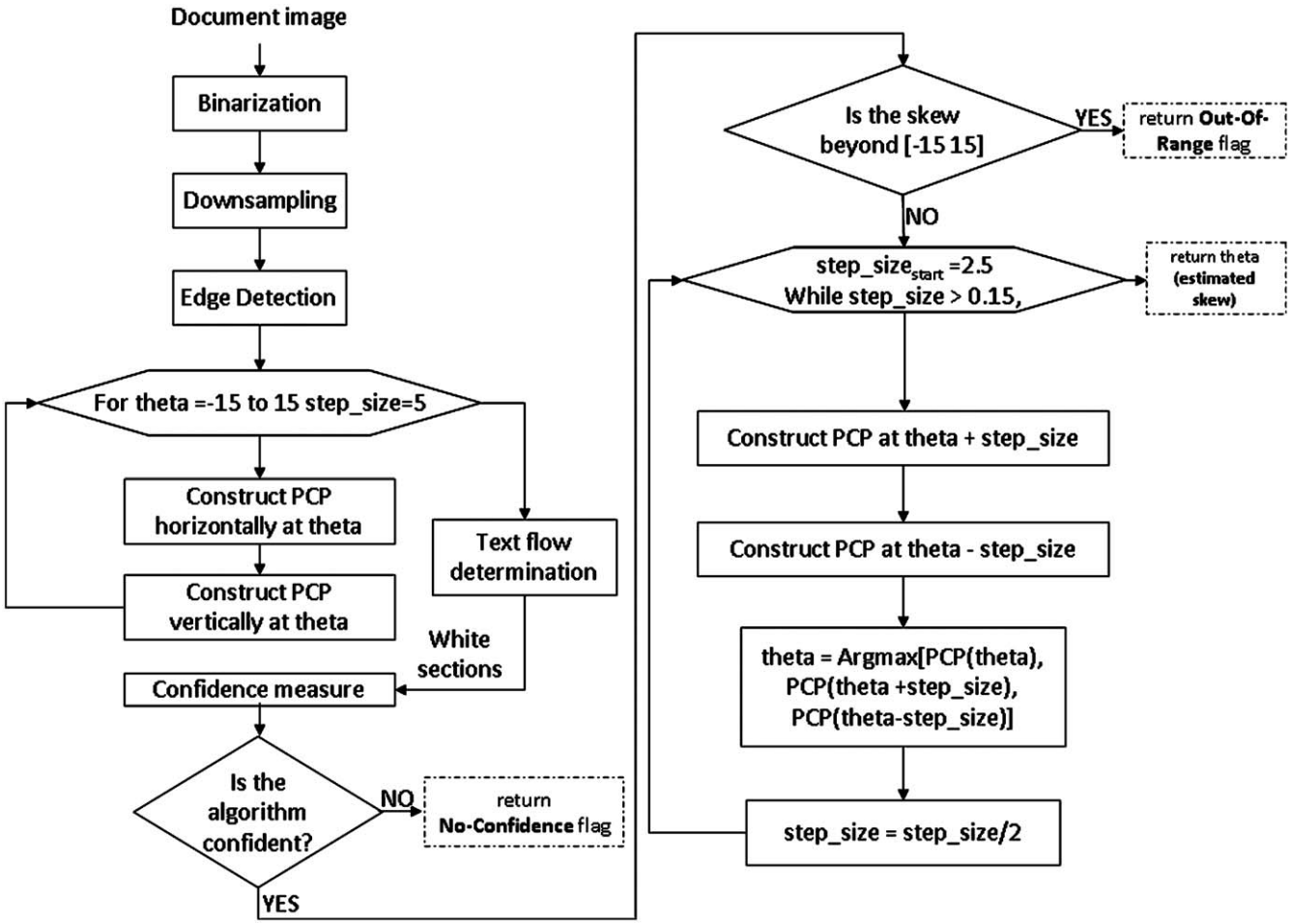


**Fig. 7.** Block diagram of the complete skew estimation process.

level estimate of skew enhances our e-PCP method in following two main aspects:

(1) Helps to determine the flow of text lines which in turn helps to obtain robust results.
(2) Reduces the number of computations during the finer level estimate of skew with an accuracy either the same or better than the PCP method [2].

Once the text flow direction is determined, the corresponding seven values of white sections will be utilized in subsequent *confidence measure* and *out-of-range-detect* modules.

### 3.2.2. Confidence measure

Along with an estimate of skew, it is always desirable to have a module which produces either a confidence measure or an estimate of probable error. The confidence value is returned to differentiate between results that are expected to be accurate and those where the method could have failed. In this work, we have two simple yet effective confidence measures for handling images of following kinds:

(1) Images without prominent maxima of white sections.
(2) Images with skew out of range (beyond $\pm 15^{\circ}$).

*Images without prominent maxima*: Sometimes in practical situations, we may encounter some images where the prominent peaks of white sections are missing. Under such circumstances, it will be better to have a mechanism with skew estimation algorithm that returns the image as such instead of yielding incorrect estimate of skew. The confidence measure module of e-PCP helps to achieve this important objective.

The values of white sections that are computed at coarse level are utilized in our measure of confidence. Using seven values of white sections that are computed at every angle from $-15^{\circ}$ to $+15^{\circ}$ in steps of $5^{\circ}$, the global maxima ($GM$) and the next local maxima ($LM$) are computed. We now use these values for devising our confidence measure as follows:

$$Diff = GM - LM \tag{2}$$

$$Confidence = \begin{cases} 1 & \text{if } Diff > T \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where $T$ is a dynamic threshold which we have empirically fixed to 10% of the global maxima. Thus confidence measure returns a value of 0 if it is likely to obtain a wrong result and a value of 1 indicates that the algorithm is confident in estimating the skew of the input document. The confidence measure is based on the fact that if white section areas for two or more angles are close to each other, it may not be a very good decision about the estimated angle.

Whenever the value 0 is returned, the subsequent steps of skew estimation are skipped and skew correction/rotation module simply displays the input image as such. In this way, confidence measure module avoids the possibility of wrongly skew correcting the image. It may be noted that the confidence measure is invoked right after

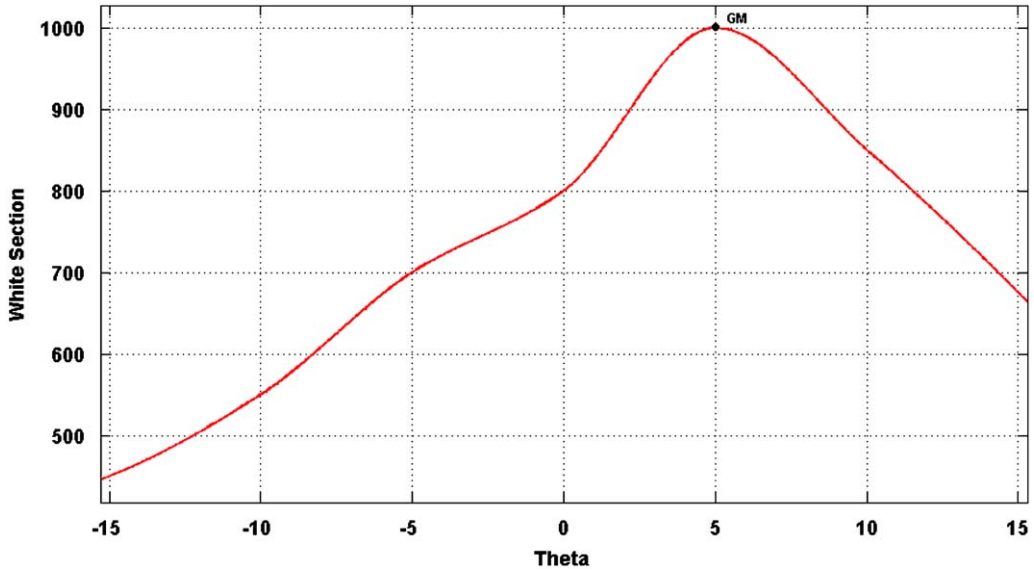


**Fig. 8.** Plot of white sections for the CONFIDENT case-1.

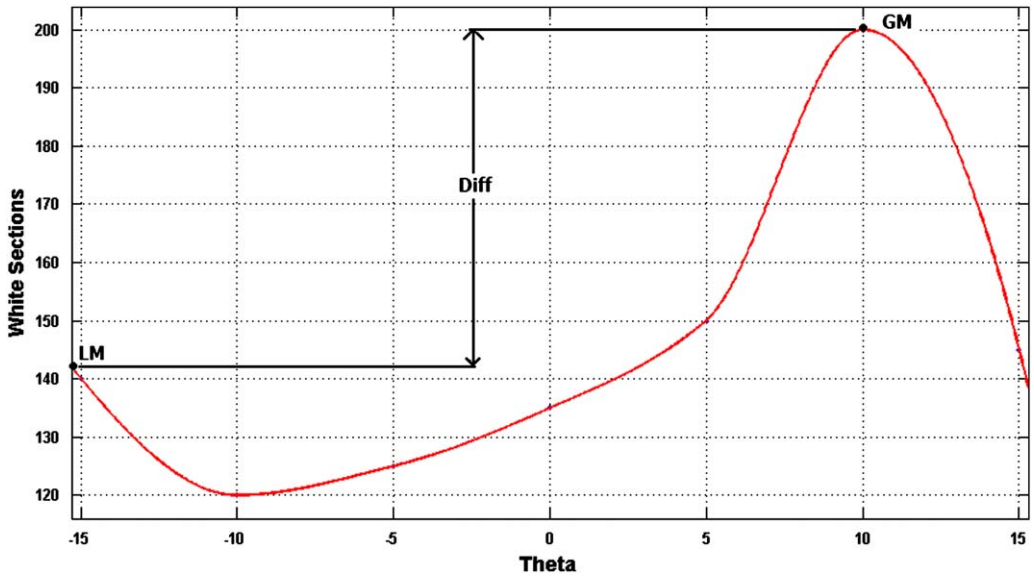


**Fig. 9.** Plot of white sections for the CONFIDENT case-2.

the coarse level estimate of the skew angle and thereby also helps to avoid subsequent computation of finer skew angle if the algorithm is not confident on a particular image.

Based on our experiments, we have ascertained that the plot of these white sections can have only any of the following three types of properties:

(1) There could be single global maxima and no local maximas as shown in Fig. 8.
(2) There could be several maximas, but difference between global and next local maxima is significant as shown in Fig. 9.
(3) There could be several maximas, but difference between global and next local maxima is very less as shown in Fig. 10.

Our confidence measure returns a value 1 on the given input image if the computed white section areas of the image for corresponding coarse angles have property like (1) and (2). On the other hand, the image is deemed as a non-confident image if the image white sections have property like (3).

Fig. 11 shows a three degree document. This is an example of a document with large scale figure and graphics, which is a very common type of image in real time situations. On this image, we observed conventional PCP [2] yielding 15°, which may be due to the fact that many vertical dividers appearing in the image are blocking the significant white section counts in the direction of actual skew angle. However, the confidence measure that we developed had low confidence on this image. This is because a plot of white sections obtained at coarse level for this image resembled the curve shown in Fig. 10, which is a no-confidence case. In this way we can say that the confidence measure of e-PCP helps to avoid getting undesired results for no-confidence images, which is a pressing need in real time applications like automated document image scanning.

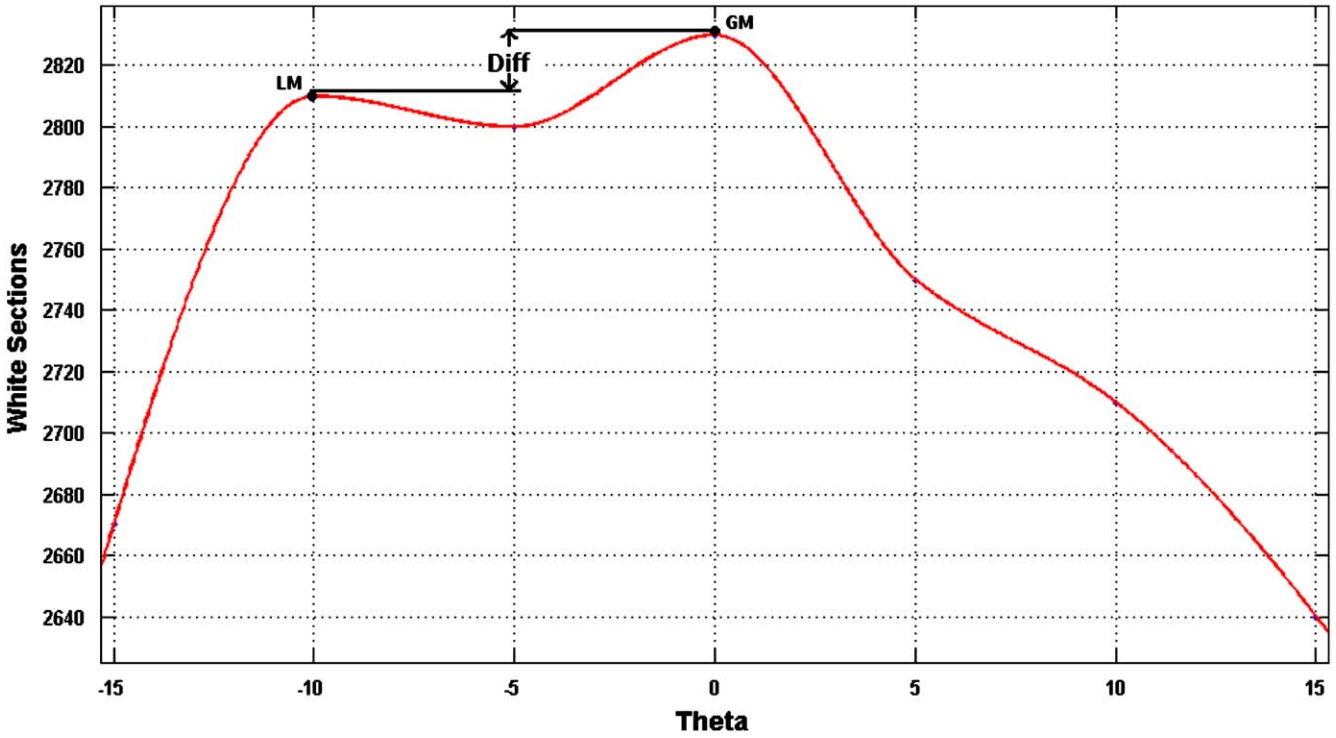


**Fig. 10.** Plot of white sections for a NOT-CONFIDENT case.

*Images with skew out of range*: For an on-platform implementation of a skew estimation algorithm, it may be desirable to have a confidence measure which detects whether the skew angle is beyond some range, for, e.g. $\pm 15^\circ$. This is because the user might deliberately scan a document with skew and does not want the subsequent deskew operation to be pursued. Based on this, we can set certain skew range upon which the skew estimation process will be carried out and can be skipped otherwise.

This section explains the procedure for detecting out of range (e.g. beyond $\pm 15^\circ$) skewed documents. It is worth mentioning that the coarse level estimate of white sections (in steps of $5^\circ$) helps us to avoid estimating the skew in the out of range documents without having the need to do whole computation. In this way, finer level estimate of skew can be totally avoided thereby reducing the number of computations.

One nice property of PCP method is that, whenever the document skew angle is beyond $\pm 15^\circ$, plot of white sections will ideally have a monotonically increasing/decreasing curve. This is illustrated in Fig. 12(a) and (b) respectively. If we encounter such a situation, i.e. the global maxima corresponding to $\pm 15^\circ$, we compute four additional white section areas at $\pm 15.5^\circ$, $\pm 16.0^\circ$, $\pm 16.5^\circ$ and $\pm 17.0^\circ$. If the values of white sections computed at these four degrees are still increasing, the algorithm returns a flag[2] indicating that skew of the input document is beyond the range of $\pm 15^\circ$.

### 3.2.3. Finer level estimate of the skew angle

From the earlier step of computation of coarse skew angle $\theta_{coarse}$, we have the computation of white section area for the scan-line angles at $-15^\circ$, $-10^\circ$, $-5^\circ$, $0^\circ$, $5^\circ$, $10^\circ$, $15^\circ$ for the correct slab orientation. In the computation of the actual maxima of white sections, we look out for the points in the coarse computation of white areas, where the samples inflect as shown in Fig. 13(a). The maxima definitely lies in the neighborhood of the inflection points. However they can only lie in left or the right sample neighborhood of the inflection point as shown in Fig. 13(a). So, in every computational step of finding the maxima, we keep the left and the right neighborhood as candidate regions where the maxima can lie (i.e. left and right interval of inflection points). Successively we bisect each region for ascertaining candidate regions for maxima and look for the same inflection points in the candidate regions, and drop the region where after next level of sampling, it is evident that no inflection point exists in that interval. This is as shown in Fig. 13(b).

For this, we have used an efficient way of converging from coarse level estimate of skew angle ($\theta_{coarse}$) to finer level skew angle estimate ($\theta^*$). The following iterative steps have been used to converge from $\theta_{coarse}$ to final skew angle $\theta^*$:

$$
\begin{aligned}
&\texttt{step\_size} = 2.5\\
&\theta^* = \theta_{coarse}\\
&\texttt{while}\ (\texttt{step\_size} \geq 0.15)\\
&\{\\
&WS_1(\theta) = \texttt{White\_Section\_Area}(\theta^*)\\
&WS_2(\theta) = \texttt{White\_Section\_Area}(\theta^* + \texttt{step\_size})\\
&WS_3(\theta) = \texttt{White\_Section\_Area}(\theta^* - \texttt{step\_size})\\
&\theta^* = \underset{\theta}{\operatorname{argmax}}(WS_1(\theta), WS_2(\theta), WS_3(\theta)).\\
&\texttt{step\_size} = \frac{step\_size}{2}\\
&\}
\end{aligned}
$$

In the above equations, *White_Section_Area*( ) refers to the process of computing the number of white sections for a particular scan-line angle. In this way, the white sections are computed at three different angles ($\theta^*, \theta^* + step\_size, \theta - step\_size$). At each iteration, $\theta^*$ gets updated with a skew angle that yielded maximum white section.

[2] In our implementations, a value of $-1$ will be returned to indicate that the skew angle of the document is beyond $\pm 15^\circ$.


**Fig. 11.** A failure and no-confidence case image for PCP and e-PCP respectively.

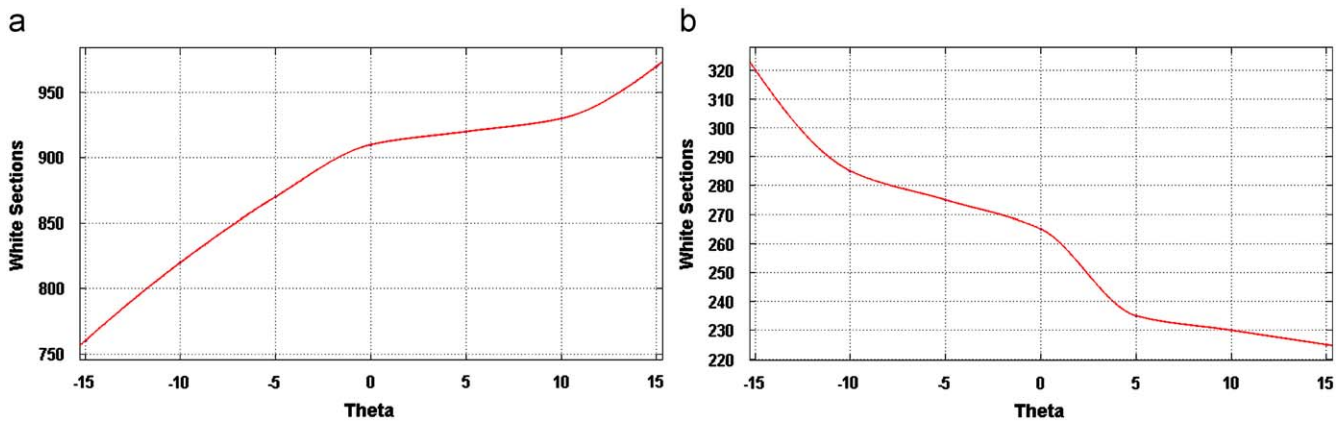


**Fig. 12.** Plot of white sections for a document skew angle more than $\pm 15^{\circ}$.

The operations inside the `while` loop are iteratively applied until the step_size becomes 0.15 (note that before the start of iteration, step_size was set to 2.5). This essentially yields the final skew angle with an accuracy of $0.05^{\circ}$ resolution.

## 4. Comment on computational load

In determining the skew angle there is trade off in the number of computations at the coarse angular values and the number of

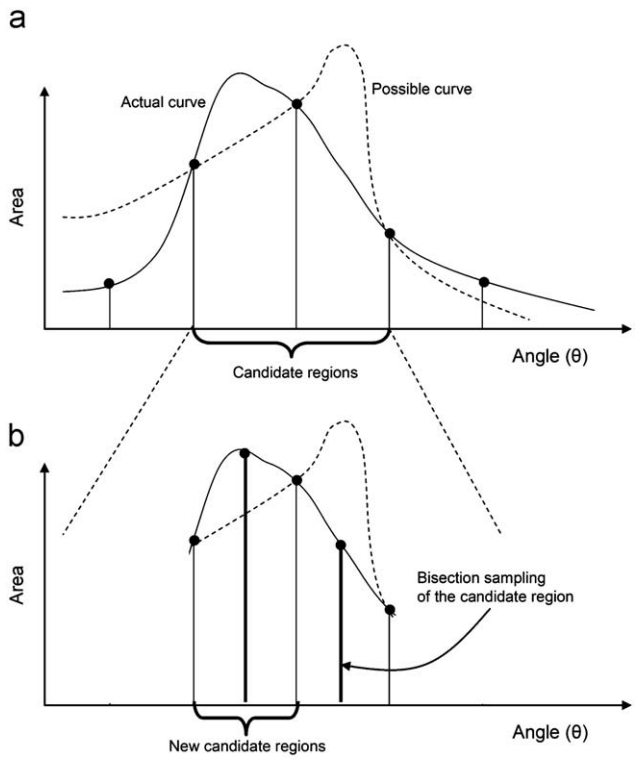


**Fig. 13.** Locating candidate regions in successive iterations.

computations at the finer angular values. If the number of coarse level calculation for determining the sense of the text flow of the document and confidence measure is high, the number of computations required at the finer levels of angles will be less to reach the required level of resolution of estimated angle. The total computation ($C$) required is the sum of the number of computations at coarse level of angles and fine level of angles. The coarse level of angles are computed at a resolution of say $\Delta$, for each of horizontal slab orientation and vertical slab orientation. The fine level computation is done by successively bisecting the $\Delta$ to reach the resolution of estimated angle to 0.15°.

Let the number of bisections required to reach the resolution of 0.15° be $n$. Assuming the skew detection range to be ±15°, the total number of computations ($C$) is

$$C = 2\left(\frac{30}{\Delta} + 1\right) + 2n \tag{4}$$

The multiplication of the first term with value 2 is due to the double computation involved during coarse-level estimate of the skew, i.e. while computing white sections for vertical and horizontal slab orientations. Similarly, the second term multiplication by value 2 is to denote the two white section computations required (towards left and right of every intermediate skew) during finer level estimate of the skew angle.

Now,

$$\frac{\Delta}{2^n} \leq 0.15 \tag{5}$$

$$\Rightarrow n = \left\lceil \log_2 \frac{\Delta}{0.15} \right\rceil \tag{6}$$

$$C = \left\lceil 2\left(\frac{30}{\Delta} + 1\right) + 2\log_2 \frac{\Delta}{0.15} \right\rceil \tag{7}$$

$$C = \left\lceil 2\left(\frac{30}{\Delta} + 1\right) + \frac{2}{\ln 2} \ln \frac{\Delta}{0.15} \right\rceil \tag{8}$$

To find the optimal $\Delta$,

$$\frac{\partial C}{\partial \Delta} = \frac{-60}{\Delta^2} + \frac{2}{\ln 2}\frac{0.15}{\Delta}\frac{1}{0.15}$$

$$\frac{\partial C}{\partial \Delta} = \frac{-60}{\Delta^2} + \frac{2}{\Delta \ln 2}$$

Setting $\partial C/\partial \Delta = 0$,

$$\frac{60}{\Delta^2} = \frac{2}{\Delta \ln 2}$$

$$\Rightarrow \Delta = 30 \ln 2 = 20.79$$

$$C|_{\Delta=20} = 20$$

$$C|_{\Delta=21} = 20$$

Therefore $\Delta = 20$ or 21 is the optimum step size during the coarse level estimate of skew angle. This results in least number of computations (i.e. $C = 20$). However for practical reasons, which requires enough sample points at coarse level of computation to determine the appropriate slab orientation/text flow, we choose $\Delta = 5°$. This results in number of computations equal to 24, which is not too far from the lowest number of computations, which is 20.

## 5. Experimental results

In this section, the performance of e-PCP method will be experimentally evaluated and compared with the conventional PCP method. It may be noted that in [2], the performance of PCP was compared with that of representative skew detection methods from the categories of PP, TC and CC based methods. It was also ascertained that the PCP method outperformed other methods for different kinds of document images. Hence we do not compare with other methods. Any better or comparable performance of ePCP over PCP method always imply a better performance of proposed method with that of PP, TC and CC based methods.

For this purpose, we used our database (HP_Labs_India_Skew_Detection_Dataset) that contains 193 real time representative skew images with following specifications:

- 30 images with vertical text lines often cutting the boundaries of the image. Few images from this category are shown in Fig. 14.
- Images with large scale tables and/or large-scale figures etc. See Fig. 15 for example images of this category.
- Images with thick side-bands and poor quality contents. Example images from this category are shown in Fig. 16.
- Images with beyond the range skew angles (i.e. ±15°).
- Images in 75, 100, 150 and 300 dpi resolutions.

In order to obtain the ground truth skew angle from the database images, we developed a software tool which returns the actual skew angle of the input document. This tool allows the user to click on two appropriate points on the image and returns the skew angle. This tool also has a mechanism of multiple checking on the same image so that the user will continue with several other pairs of points unless he/she is satisfied. Finally, mode of the accepted skew angles obtained through multiple pairs of points will be deemed as the actual (or ground truth) skew angle of the document. In all our experiments of PCP and e-PCP, we restricted the maximum skew range search to ±15°. We run both PCP and e-PCP on this image database and results are as shown in Table 3.

Table 3 compares PCP and e-PCP methods with important evaluating parameters such as average error, running time, computational load and measure of confidence. The average error is defined as the

average of the absolute deviation of the computed skew angle from the ground truth skew angle:

$$Average\ Error = \frac{\sum_{i=1}^{N} abs(\bar{\theta} - \theta^*)}{N} \quad (9)$$

where $\bar{\theta}$ is the ground truth skew angle, $\theta^*$ is the computed skew angle and $N$ is the number of images in the database.

From the average error in the table, we can conclude that the e-PCP method yields more accurate estimate of skew angle.

The computation of white sections for various angles is the most expensive part of both PCP and e-PCP algorithms, and it is evident from the table that the number of PCP computation steps is significantly reduced in case of e-PCP algorithm compared to the PCP algorithm [2]. Thus the e-PCP algorithm achieves efficient results both in terms of accuracy and reduced running time. Table 4 shows the results of PCP and e-PCP methods on 30 VFT documents that we created. From Table 4, it is very clear that PCP often leads to erroneous results due to fixed vertical slab orientation. This vertical slab orientation often does not yield sufficient white sections for PCP to make a decision. Since e-PCP has a mechanism for choosing slab orientation based on the text flow direction, it made use of horizontal slab and thus yielded more accurate results.

**Fig. 14.** Example images of Chinese images with text lines cutting boundaries of document.

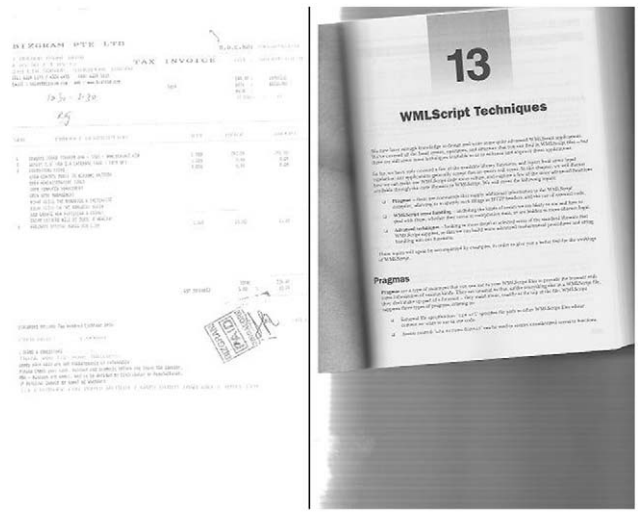

**Fig. 16.** Example images of poor quality content and dark side-band images.

**Table 3**
Evaluating PCP and e-PCP methods.

| Evaluating parameter | PCP | e-PCP |
|---|---|---|
| Average error | 0.5744 | 0.3580 |
| Number of PCP computations | 38 | 24 |
| Running time (s) | 0.41 | 0.29 |
| Confidence measure | NA | Robust |
| Add-on | NA | Detects text flow |

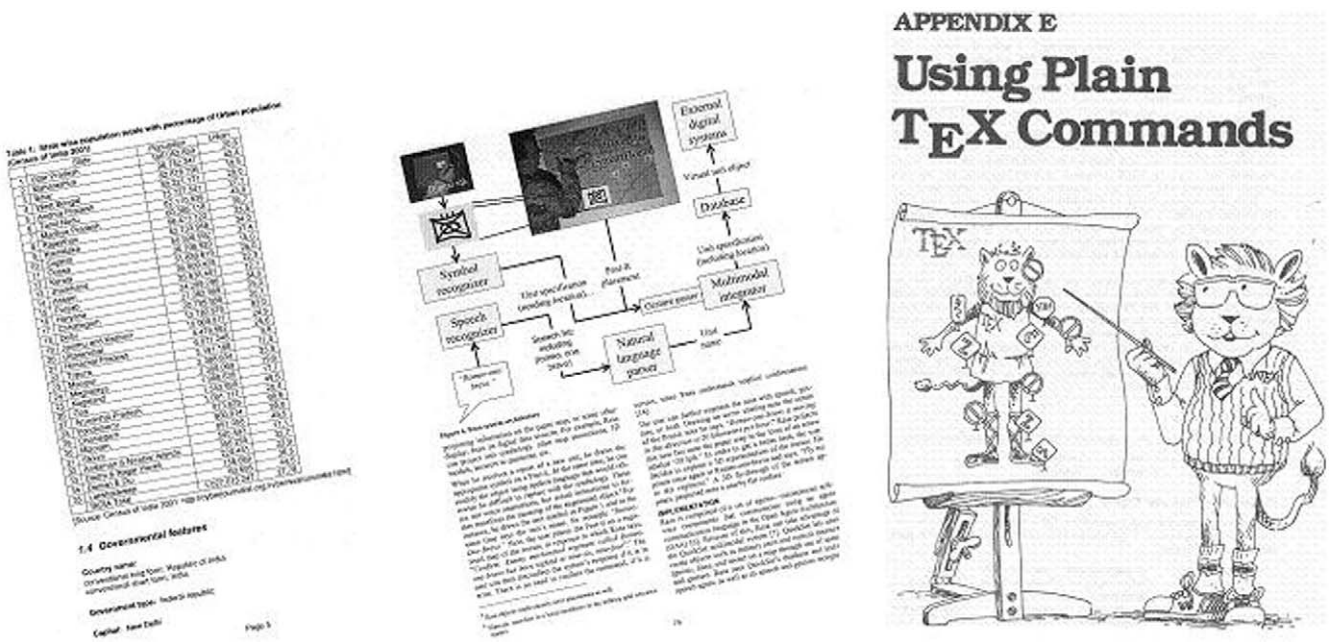

**Fig. 15.** Example images of large scale tables and figures.

**Table 4**
Skew angle obtained by PCP and e-PCP on VFT documents.

| Image no. | Ground truth (deg) | PCP | e-PCP |
|---|---|---|---|
| 1 | −12 | 9.7 | −11.75 |
| 2 | −6 | −14.9 | −5.9 |
| 3 | 12 | 14.9 | 12.35 |
| 4 | 6 | 15 | 6.1 |
| 5 | −6 | −10.5 | −5.9 |
| 6 | 12 | 0.1 | 11.95 |
| 7 | 6 | 15 | 5.75 |
| 8 | 12 | 15 | 11.75 |
| 9 | 6 | 15 | 5.75 |
| 10 | 2.5 | 1.9 | 2.3 |
| 11 | −4 | −15 | 5.75 |
| 12 | −6.5 | −14.9 | −6.25 |
| 13 | 9 | 14.9 | 8.55 |
| 14 | 0 | 0.1 | 0 |
| 15 | 0 | 15 | 0 |
| 16 | 3 | 14.9 | 3 |
| 17 | −3 | 15 | −2.85 |
| 18 | 10 | 14.9 | 10.9 |
| 19 | 0 | −14.9 | 0 |
| 20 | −8 | −14.9 | −7.85 |
| 21 | −9 | 15 | −9.3 |
| 22 | 6 | 6.7 | 6.05 |
| 23 | 12 | 12.7 | 11.8 |
| 24 | −6 | −6.7 | −5.9 |
| 25 | −12 | −12.3 | −12.15 |
| 26 | 0 | 14.9 | 0 |
| 27 | 6 | 14.9 | 0 |
| 28 | 6 | 14.9 | 6.1 |
| 29 | −6 | 0.1 | −5.9 |
| 30 | 12 | 14.9 | −12.15 |

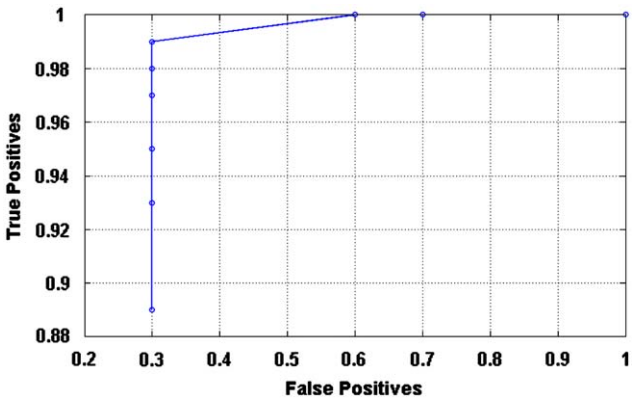


**Fig. 17.** ROC curve of the confidence measure.

The confidence measure of the e-PCP algorithm is robust. Fig. 17 shows the plot of ROC curve for the confidence measure for changing threshold which is the difference between primary maxima and secondary maxima. As can be seen from the ROC curve, it resembles the case where the area under the curve is close to 1 which indicates that the confidence measure is good.

Another set of experiments were conducted on the database created by the authors of original PCP, Chou et al. [2]. This database contains five categories of documents: (i) English documents, (ii) Chinese and Japanese documents, (iii) documents with large scale figures, (iv) documents with forms or tables and (v) multilingual documents. Each category contains 100 images of 20 documents with each being skewed by five ground truth skew angles (0°, 6°, 12°, −6° and −12°). For our experiments, we considered images of categories (iii), (iv) and (v) only since we had enough samples for categories (i) and (ii) in our database of 193 images. Table 5 shows the average error obtained by PCP and e-PCP methods on these chosen categories of images. This is yet another evidence in favor of e-PCP method since it obtains better or comparable results with that of PCP with much reduced computational load.

**Table 5**
Average error obtained on different categories of images.

| Image category | PCP | e-PCP |
|---|---|---|
| Documents with large scale figures | 0.4781 | 0.4290 |
| Documents with forms and tables | 0.2659 | 0.2546 |
| Multiscript documents | 0.1953 | 0.1946 |

## 6. Conclusions

In this paper, we presented e-PCP which is a robust skew estimation algorithm. This algorithm is an enhancement to the PCP algorithm [2], a very recent and robust algorithm in literature. The e-PCP algorithm enhanced the PCP algorithm with respect to the following important issues:

- Robust results when there are vertical text lines touching the boundaries of the document. This is also the case where PCP algorithm often fails.
- The proposed method is insensitive to the slab width, especially when there are vertical text lines, unlike the PCP algorithm.
- In case of e-PCP, there is a significant reduction in number of white section computation steps (refer Table 3), which is the most costly operation in the whole skew estimation process. In this way, e-PCP helps in speeding up the whole skew estimation process.
- The e-PCP algorithm has a robust confidence measure, which often is of great importance in real time automated systems to filter cases where skew estimation is not confident, instead of outputting a wrong result. This in turn helps the e-PCP algorithm to be deployed for real time applications.

We conducted extensive set of experiments on databases containing complex documents and ascertained the efficacy of the e-PCP algorithm. Based on the results we conclude that with reduced number of computations, the performance of e-PCP is either the same or better than PCP for documents containing text, text with large scale figures, text with large tables and multi-script documents. However, the accuracy of e-PCP is significantly better for VFT documents, where often PCP fails to perform due to lack of sufficient white sections.

## Acknowledgments

We would like to thank C.-H. Chou et al., authors of PCP [2], for providing us their database of skewed documents for this study.

## Appendix A. Accelerating the execution time of the e-PCP algorithm

In this section, we describe a faster way to accelerate the execution speed of the e-PCP algorithm. For this purpose, we have developed the whole algorithm by using following imporant strategies:

(1) Fixed point (integer-only) implementation.
(2) Using Look-up tables by pre-computing repetetive/expensive operations.

### *A.1. Fixed point implementation*

The fixed point implementation not only enhances the speed but is also a prime requirement for embedding the algorithm into hardware products.

There are several floating point operations involved for the realization of the e-PCP skew estimation algorithm, but we have implemented all of them in an equivalent fixed point operations without loosing any accuracy. The following shows some prominent float operations involved and our equivalent fixed point implementation of the same:

(1) *ceil*$(x,y)$: The equivalent fixed point command is $(x+y-1)/y$.
(2) *round*$(x,y)$: The equivalent fixed point command is $(2x+1)/2y$.

### *A.2. Using look-up tables by pre-computing repetitive/expensive operations*

The most repetitive/expensive calculation involved in the implementation of e-PCP algorithm is the computation of the $x$ and $y$ coordinate values to navigate in a particular scan-line angle. This is computed for every new pixel while scanning either row-wise or column-wise.

For this, we have made use of look-up table (matrix) by precomputing the offset values (new y-coordinate values) for every angle in the given skew range (0–45°). This requires a 2D static memory for storing these offset values in the lookup matrix. This is the most significant part of memory consumption by this algorithm. Note the inherent advantage of storing offset values for positive range—it allowed to store the offset values using unsigned type (for e.g.: uint8) as opposed to signed type (for e.g.: sint16) which also helped to reduce the static memory usage by 50%.

These offset values are pre-computed assuming (0, 0) as the origin. We only add these off-set values to the $y$-coordinate of corresponding pixel in each row, in order to navigate in a particular scan-line angle. For every scan-line angle, the offset values are precomputed for 150 pixels in order to cope up with maximum slab width of 150. For optimal results, we empirically fixed a value of 150 to the slab width.

In our implementations we have a lookup matrix called `lookup_finer[][]` of size $900 \times 150$. This matrix stores the offset values of each scan-line angle from 0 to +45° in steps of 0.05° and hence the number of rows is 900 and 150 represents the maximum slab-width that is used in our implementation, which can always be extended. Note that, a row in the lookup matrix corresponds to offset values of a particular scan-line angle. The `lookup_finer[][]` matrix values are used during following three main operations:

(1) coarse level estimate of the skew angle,
(2) out-of-range detect and
(3) coarse-to-fine level skew estimation.

Note that for negative angles, we simply negate the offset values of the corresponding positive angles in the lookup table.

The following algorithm computes the coordinate values to traverse in a particular scan-line angle from left to right.

```
Count = 0;
for y = 0 : no_of_rows,
{
  y_origin = y;
  while(x <= no_of_columns)
  {
    new_x = x++;
    new_y = y_offset[Count++] * sign + y_origin;
  }
}
```

where `new_x` and `new_y` are the new $x$ and $y$ coordinates computed to traverse in a particular scan-line angle in the image. The vector `y_offset` is a pointer to a row corresponding to the scan-line angle in the lookup table `(Lookup_finer[][])`. The value of the variable `sign` will be +1 for positive angles and −1 otherwise. Note that the above algorithm only computes the new coordinate values to traverse in a particular scan-line angle. Other steps such as skipping a current section (if the current pixel happens to be black) or continuing with the same section (if the current pixel is white) are all obvious and hence not included in the algorithm. Note that this strategy to speed up the computations can also be used in the existing PCP [2] algorithm.

## References

[1] A. Hashizume, P.S. Yeh, A. Rosenfeld, A method of detecting the orientation of aligned components, Pattern Recognition Letters 4 (1986) 125–132.
[2] C.-H. Chou, S.-Y. Chu, F. Chang, Estimation of skew angles for scanned documents based on piecewise covering by parallelograms, Pattern Recognition 40 (2007) 443–455.
[3] E. Kavallieratou, N. Fakotakis, G. Kokkinakis, Skew angle estimation for printed and handwritten documents using Wigner–Viller distribution, Image and Vision Computing 20 (2002) 813–824.
[4] H.F. Jiang, C.C. Han, K.C. Fan, A fast approach to the detection and correction of skew documents, Pattern Recognition Letters 18 (7) (1997) 675–686.
[5] H. Yan, Skew correction of document images using interline cross-correlation, CVGIP—Graphical Models and Image Processing 55 (6) (1993) 538–543.
[6] S. Li, Q. Shen, J. Sun, Skew detection using wavelet decomposition and projection profile analysis, Pattern Recognition Letters 28 (2007) 555–562.
[7] C. Singh, N. Bhatia, A. Kaur, Hough transform based fast skew detection and accurate skew correction methods, Pattern Recognition 41 (2008) 3528–3546.
[8] T. Akiyama, N. Hagita, Automated entry system for printed documents, Pattern Recognition 23 (1990) 1141–1154.
[9] U. Pal, B.B. Chaudhuri, An improved document skew angle estimation technique, Pattern Recognition Letters 17 (8) (1996) 899–904.
[10] X. Jiang, H. Bunke, D. Widemer-Kljajo, Skew detection of document images by focused nearest-neighbor clustering, in: Proceedings of the Fifth International Conference on Document Analysis and Recognition, 1999, pp. 629–632.
[11] Y. Ishitani, Document skew detection based on local region complexity, in: Proceedings of the Second International Conference on Document Analysis and Recognition, 1993, pp. 49–52.
[12] C. Singh, N. Bhatia, A. Kaur, Hough transform based fast skew detection and accurate skew correction methods, Pattern Recognition 41 (2008) 3528–3546.

**About the Author**—PRASENJIT DEY is a research scientist at HP Labs India. He received his Ph.D. (2004) in Communications Engineering from Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, on an EPFL Doctoral School Fellowship Award and an M.Tech. (1998) in Communications Engineering from IIT, Delhi, India, on a Samsung Fellowship Award. He worked for Sasken Communication Technologies, Bangalore, India as a Software Engineer and as an Intern for Samsung Research Laboratories, Seoul, South Korea. His current interests are in the areas of information theory, signal processing and pattern recognition in the context of multimodal systems. He is a member of IEEE and ACM.

**About the Author**—NOUSHATH S is a research consultant to HP Labs, India. He completed his Ph.D. (2008) in Computer Science from the University of Mysore, India. His areas of research interests include document analysis and application of subspace algorithms for image recognition. He has over 10 publications in all areas of interest. His other interest is in playing carrom board (twice inter-collegiate champion) and he has ardent music liking for A.R. Rahman's songs.