Estimation of Tilt in Characters and Correction for better Readability by OCR Systems

C. S. Vijayashree
P.E.T Research Centre,
PES College of Engg.
Mandya, India-571401

Vishwanath C. Kagawade
Basaveswara College of Engg.
Bagalkot
India

T. Vasudev
Maharaja Research Foundation
MIT Campus, Belawadi,
S R Patna, India-571438

ABSTRACT
The existing Optical Character Readers (OCRs) are capable of reading linear form text and have limitations to read artistic and non-linear form text. The tilt in characters contributes a major share in affecting the efficiency of the recognition algorithms. This paper presents a technique to estimate and correct the vertical tilt in printed characters of English in order to make an OCR to read the text more efficiently. The input characters are assumed to be segmented from the document image and free from noise. Initially, the direction of tilt of the characters is detected using a heuristically constructed knowledge base. Next, the inclination of the character to its base is estimated using line drawing algorithm. Finally, the estimated tilt is corrected through rotation in counter direction of the tilt. The method has been tested with sufficient samples and readability analysis is performed with an OCR. Experimental results show an average improvement in readability by OCR from 20% before tilt correction to 82% after the tilt correction.

Keywords
Linear text, Artistic text, Tilt in characters, Tilt correction, OCR.

1. INTRODUCTION
A significant area in the field of Digital Image Processing is Document Image Analysis (DIA). DIA is very important in applications like document identification/recognition, language identification, automatic reading from document etc. Many researchers are working on different problems on document images starting from image acquisition to image understanding [1,2]. Processing activities in DIA can be divided into Pre-processing, Segmentation, Script Identification, Page Layout Analysis (PLA) and Classification, Character Recognition etc [3], and these have lead into many vibrant research problems [2]. The results of the research on the above problems are gradually converging towards generic solutions to major issues in DIA.

In spite of considerable research work in the area of DIA, a major issue which is not sufficiently addressed is the detection and correction of skew or tilt in characters. Tilt is the angular slant to the baseline introduced in the character. Tilted characters are mainly noticed in many artistic texts. Fig. 1 shows few samples of text with tilted characters. The characters extracted from such artistic text exhibit inherent tilt considerably. Fig.2 show few examples of tilted characters segmented from artistic text. Such tilted characters hinder the investigation of generic methods of recognition and the efficiency of recognition drops relatively. Hence tilt in characters contributes a major share in affecting the efficiency of the recognition algorithms.

Literature survey reveals that most of the character recognition algorithms assume that the input is tilt corrected. When tilted characters are subjected to recognition through any existing algorithms, the rate of recognition obviously becomes low. The rate of character recognition is inversely proportional to the degree of tilt in characters i.e. higher the degree of tilt, lower is the recognition rate [4-6].

Considerable amount of research is reported in literature on the skew detection [7-13] and correction of document images. The document skew detection and correction algorithms cannot be extended to detect and correct tilt in characters. The characteristics considered for detection of skew at document level is different from the characteristics considered for skew detection at character level. Generally, global characteristics of the document like finding line orientations using Hough transformations [9], slope between nearest-neighbor chain (NNC) obtained in the documents [13], horizontal
The characters extracted from any artistic form text possess an implicit tilt [14]. Due to this OCRs fail considerably to read such tilted characters. In this research work, we propose a methodology to estimate and correct the vertical tilt of printed characters of English subjected for recognition. The input to the method is the segmented characters obtained from an artistic text through connected component labeling [15,16]. The proposed method initially performs a macro level decision to find the direction of tilt i.e. detection of tilt is towards left or right from baseline. The proposed tilt detection algorithm detects the direction of tilt in the input character with the support of a heuristic knowledgebase. Next, the degree of tilt to the baseline is estimated using line drawing algorithm. Finally the character is rotated by the estimated tilt in the counter direction of the tilt.

The block diagram shown in Fig.3 indicates the sequence of different processing stages performed in the proposed system. Input to the procedure is a character which is segmented from a document image containing artistic text.

The rest of the paper is organized as follows: The detection of the direction of tilt in characters is explained in section 2. Tilt angle estimation and correction of tilt in the input character is explained in section 3. Section 4 explains the methodology used. Experimental results are illustrated in section 5. Section 6 explains some of the limitations scenarios and conclusion is given in section 7.

Table 1. Rules for Tilt Direction Detection

<table>
<thead>
<tr>
<th>Slno.</th>
<th>Presence of a part of the Character</th>
<th>Tilt</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Left Top</td>
<td>Right Top</td>
<td>Left Bottom</td>
</tr>
<tr>
<td>1</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>2</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>3</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>4</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>5</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>6</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>7</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>8</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>9</td>
<td>N</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>10</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>11</td>
<td>Y</td>
<td>Y</td>
<td>-</td>
</tr>
<tr>
<td>12</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>13</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
</tbody>
</table>
In case of conflicts [a] through [d], a next level decision is carried out to detect tilt based on the density of the pixels in the corner areas of the rectangle box. The pixel densities in the four corners are computed as left top density (LTD), right top density (RTD), left bottom density (LBD) and right bottom density (RBD). A decision rule is devised as given in Table 2 to make the second level decision to detect tilt direction for conflict cases [a] through [d] of Table 1.

The conflict case [e] arises when no part of the character is found in any of the corner areas of the rectangular area enclosing the character as shown in Fig. 6(a). In order to handle such conflicts, another set of decision rules are applied to detect the direction of tilt. The rectangular area defining the character is divided into 4 equal quadrants as in Fig. 6(b). The decision on direction of tilt is then based on the position of the edge of the character touching the outer boundary in the 4 quadrants. The outer edges of the quadrants formed in the rectangular boundary enclosing the character are named as illustrated in Fig. 6(b).

The outer edges labelled $h_k$ and $v_k$ represent horizontal and vertical edges respectively of $k^{th}$ quadrant. $P_k$ represents a point on the edge in $k^{th}$ quadrant. An operator $\delta$ is defined in this method to find a point of the character touching an edge of the bounding box. The notation $(P_k \delta v_k)$ indicates the point at which character touches the edge $v_k$. The decision rules for tilt direction detection in conflict case [e] is illustrated in Table 3.

The knowledgebase constructed is used as a supporting system to enable a macro level detection of direction of tilt. As the knowledgebase is constructed based on the structure of English characters, experimentation is carried out on good number of tilted English characters. In general, the developed model provides optimum results for tilt angles between 5° - 45°. The method is tested for both upper case and lower case characters. When the tilt angle is outside this range, the corner area triangle methodology does not provide definite results. For the English characters with tilt between 5° - 45°, the success rate is around 93%. Once the direction of tilt is detected the character is subjected for estimation of tilt angle. The procedure for estimating the angle of tilt in the character is explained in the subsequent section.

### Table 3. Decision Rules for Tilt Direction Detection

<table>
<thead>
<tr>
<th>Conflict Case</th>
<th>Condition</th>
<th>Tilt Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>RTD &gt; LTD</td>
<td>Left</td>
</tr>
<tr>
<td></td>
<td>(RBD &gt; RTD) and (RBD &gt; LTD)</td>
<td>Left</td>
</tr>
<tr>
<td>b</td>
<td>LTD &gt; RTD</td>
<td>Right</td>
</tr>
<tr>
<td></td>
<td>(LBD &gt; RTD) and (LBD &gt; LTD)</td>
<td>Right</td>
</tr>
<tr>
<td>c</td>
<td>RBD &gt; LBD</td>
<td>Right</td>
</tr>
<tr>
<td>d</td>
<td>RTD &gt; LTD</td>
<td>Left</td>
</tr>
<tr>
<td></td>
<td>(LTD &gt; RTD) and (LTD &lt; RBD)</td>
<td>Left</td>
</tr>
</tbody>
</table>

The procedure for estimating the angle of tilt in the character is described as follows:

#### 3. TILT ANGLE ESTIMATION AND CORRECTION

Once the direction of tilt is obtained, the angle of tilt in that direction is to be estimated. A number of tilted English characters are used in this section for the estimation of tilt angle. For any tilted character, the baseline is defined, which is a line parallel to the horizontal line and passing through the center of gravity of the character. The angle of tilt is calculated in relation to the baseline.

The procedure for estimating the angle of tilt in the character is described as follows:

The DDA line drawing algorithm is used to draw a line between two specified points. A series of lines are drawn and a specific line is selected out of them. The selected line is an approximation of the tilt of the character to the baseline. Given below is the procedure for estimating the angle of tilt for a left tilted character.

#### 3.1. Estimation Algorithm

1. **Select Line**
   - Select the line with the maximum negative slope among all the lines drawn.

2. **Calculate Tilt Angle**
   - The line is approximated to be a straight line.
   - The angle of tilt is calculated using the slope of the line and the baseline.

**Algorithm Steps**

1. **Initialization**
   - Let $x_1, y_1$ be the starting point of the line.
   - Let $x_2, y_2$ be the ending point of the line.
   - Let $m$ be the slope of the line.

2. **Calculate Tilt Angle**
   - Compute the angle of tilt $\theta$ using the formula:
     \[ \theta = \arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \]

3. **Output Tilt Angle**
   - The calculated angle $\theta$ is the estimated angle of tilt.

**Example**

Given a character with a baseline through its center of gravity, the line drawn through the center with the maximum negative slope is selected. The angle of tilt is estimated using the slope of this line and the baseline.

**Fig 5** show few sample characters used for detection of tilt directions based on heuristic decision rules specified in this section.

**Fig 5**: Samples used in Tilt Direction Detection
1. In the rectangle enclosing the character, a point (A) is identified. The point A is obtained through a search upwards starting from the mid point of the vertical boundary on the side of the tilt. A is the first point that touches the boundary line and the same is shown in Fig. 7.

![Fig 7: Identifying the Initial Point of the Line](image)

2. A point B is searched on the baseline starting from the tilt side corner point till a line is constructed that just touches the character. This procedure is illustrated in Fig. 8. The points B₀, B₁, B₂, Bₙ₋₁, Bₙ are the series of points searched for drawing the lines AB₀, AB₁ etc till the line drawn just touches the character. The point B₀ is the first point on the baseline starting from the side of the tilt. Bₙ is the final point searched on baseline and the line ABₙ is the required line that touches the character. The final point Bₙ on the baseline obtained is denoted as B.

![Fig. 8 Illustration of Drawing Line AB to Detect the Angle of Tilt.](image)

3. The line AB is used to identify the tilt of the character. The angle formed by line AB to the baseline gives the angle of tilt θ of the character. Fig. 9 illustrates the line AB and the angle of tilt detected.

![Fig 9: Line AB Used to Detect the angle of Tilt.](image)

4. Next the character is rotated by 90 minus θ in the counter direction of tilt. Fig. 10 illustrates the tilt corrected character. Fig. 11 shows some samples of tilt corrected characters.

![Fig 10: Tilt Corrected Character](image)

![Fig 11: Characters after Tilt Angle Correction of Samples Shown in Fig. 5.](image)

4. METHODOLOGY

The complete methodology used in tilt detection and correction is explained below in the form of an algorithm. The same is illustrated in Fig. 12.

1. Input character segmented from an artistic text
2. Identify the tilt direction using the knowledgebase and the rules for detecting the tilt as given in section 2
3. If the character is not tilted, exit
4. If the character is left tilted
   a. Search the points A and B through the procedure explained in Section 3
   b. Draw the logical line L₁ from A to B
   c. Find the angle θ between L₁ and baseline. This is the estimated angle of tilt in the character
   d. Rotate the character towards right by 90 - θ degrees
   e. Exit
5. If the character is right tilted
   a. Search the points A and B through the procedure explained in Section 3
   b. Draw the logical line L₁ from A to B
   c. Find the angle θ between L₁ and baseline. This is the estimated angle of tilt in the character
   d. Rotate the character towards left by 90 - θ degrees
   e. Exit
5. EXPERIMENTAL RESULTS

The proposed method has been implemented in the MATLAB R2009a. Characters with different tilt are considered for testing. Optimum results have been obtained for characters within 45° left or right tilt. The impact of implementation of tilt correction on characters is to increase the readability of OCR. Analysis of readability by an OCR before transformation, and after tilt correction is performed with respect to English text using the OCR “Readiris Pro 9” (http://www.irislink.com). Some of the samples tried and the results obtained are illustrated in the below Table 4.

As illustrated in Table 4, the readability of OCR is improved considerably after tilt correction of the characters implying existing OCRs are not suitable to read tilted characters. The figures of the results of tilt direction detection and estimation for tilt angle are provided in Table 5 and Table 6.

Table 4: OCR Readability Results Before and After Tilt Correction of Character.

<table>
<thead>
<tr>
<th>Input Character</th>
<th>Readiris output</th>
<th>Corrected Character</th>
<th>Readiris output</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>I</td>
<td>B</td>
<td>B</td>
</tr>
<tr>
<td>C</td>
<td>E</td>
<td>C</td>
<td>C</td>
</tr>
<tr>
<td>E</td>
<td>C</td>
<td>E</td>
<td>E</td>
</tr>
<tr>
<td>C</td>
<td>E</td>
<td>C</td>
<td>C</td>
</tr>
<tr>
<td>E</td>
<td>C</td>
<td>E</td>
<td>E</td>
</tr>
</tbody>
</table>

Table 5 provides the results for detection of the direction of tilt. An overall 82.31% of correct tilt direction is detected with 13.97% of incorrect detection and 3.72% of failures through the proposed method. Incorrect tilt direction detection is noticed when the tilt direction cannot be detected using the proposed algorithm. This happens in the samples when the tilt is outside the range 5° - 45°. Failure to detect tilt direction is also noticed when the character font styles do not fit into the decision rules formulated.

Table 6 provides the results for the estimation of the angle of tilt for the input characters that are correctly identified the direction of tilt. The proposed method to estimate the angle of tilt based on detection of tilt direction shows an overall correct tilt angle estimation of 82.25% with 8.96% over estimation and 8.79% under estimation. Correct estimation is noted when the character is aligned to 0° on rotation by 90° - θ in the counter direction of tilt. Under estimation of angle of tilt is noted when counter rotation by 90° - θ still exhibits some residual tilt in the character. Over estimation of angle of tilt is noted when counter rotation by 90° - θ exhibits tilt in the direction opposite to the original direction of tilt.

Table 5: Results Obtained during Tilt Direction Detection

<table>
<thead>
<tr>
<th>No. of Samples</th>
<th>No. of correct Detection</th>
<th>No. of Incorrect Detection</th>
<th>Failures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left Tilt</td>
<td>490</td>
<td>396</td>
<td>73</td>
</tr>
<tr>
<td>Right Tilt</td>
<td>468</td>
<td>377</td>
<td>67</td>
</tr>
<tr>
<td>Without Tilt</td>
<td>252</td>
<td>223</td>
<td>29</td>
</tr>
<tr>
<td>Total</td>
<td>1210</td>
<td>996 (82.31%)</td>
<td>169 (13.97%)</td>
</tr>
</tbody>
</table>

Table 6: Results Obtained During Tilt Angle Estimation

<table>
<thead>
<tr>
<th>No. of samples</th>
<th>No. of correct estimation</th>
<th>No. of over estimation</th>
<th>No. of under estimation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left Tilt</td>
<td>396</td>
<td>328</td>
<td>32</td>
</tr>
<tr>
<td>Right Tilt</td>
<td>252</td>
<td>205</td>
<td>26</td>
</tr>
<tr>
<td>Total</td>
<td>648</td>
<td>533 (82.25%)</td>
<td>58 (8.96%)</td>
</tr>
</tbody>
</table>

6. LIMITATIONS SCENARIOS

There are few instances of tilt scenarios which show error in the current methodology proposed in the previous sections. These failures scenarios are noticed in Tilt direction detection and not in the Tilt angle calculation procedures.

The errors in tilt correction are noticed due to angle of tilt, the character considered and the font style of the character. The most optimum tilt direction detection is obtained when the tilt angle is between 5° and 45°. When the angle is less than 5° or greater than 45°, the tilt direction detection is defective. This is because the logic based on the presence of character in the corner rectangles proves ineffective. For some characters, the tilt direction detection procedure does not work due to the character’s inherent style, for example y, A, v. Similarly some font styles do not lend themselves to the corner triangle tilt direction detection method. The tilt angle calculation method provides inaccurate results for some characters depending on the style of the character. This can be observed for some characters like T, f and d in specific font styles. In some font styles characters like T lend themselves to overestimation of angle of tilt and characters like d and f lend themselves to
under estimation of the angle of tilt. Fig. 13 illustrates few instances of failures in tilt direction detection. Fig. 14 illustrates few instances of over estimation of tilt angle. Fig. 15 illustrates few instances of under estimation of tilt angle.

7. CONCLUSION
In this paper, a method to estimate and correct the tilt in English printed text is presented which are mainly noticed in artistic text. The method is based on line drawing algorithm. The experimental results exhibit an average readability of 82% by an OCR. However, the approach has certain limitations in detecting the tilt direction and estimating the angle of tilt exactly in some cases. The failures are mainly due to fault in detection of tilt direction. However, the approach is fairly good in estimating the angle of tilt in case of correct detection of tile direction. The proposed method is not suitable for some characters like A, V etc in some specific font styles. It is also not suitable for characters tilted at more than 45°. A method is under investigation to perform recursive approach to re-estimate over and under estimation of tilt angles. Further, there is much scope for research in detection of tilt direction through creating a much stronger knowledgebase. The work can be further continued to detect tilt especially for characters with large angle of tilt and handwritten characters.

8. ACKNOWLEDGEMENT
The authors thank the Managements and administrations of Maharaja Research Foundation, Mysore and PET Research Foundation, Mandy for their support and encouragement extended towards the research.

9. REFERENCES
[12] Shivakumar P., 2005, Generation of Complete Large Document Images from Split Components, Ph.D thesis under the supervision of Hemanthakumar G, University of Mysore, India
Applications, Vol. 7 No. 3.


