Slant Correction in an Optimization Framework

S. Uchida
Faculty of Information Science and Electrical Engineering
Kyushu University, Japan
In this presentation ....

**Topic 1**
Non-uniform slant correction for handwritten word

and

**Topic 2** (planning phase)
Non-uniform skew correction for scanned document
Topic 1

Non-uniform slant correction for handwritten word
Slant correction of handwritten word

Handwritten word

Slant-corrected word
Conventional techniques

- **Averaging**
  - Masking
  - Local slants
  - Averaged slant

- **Projection histograms**

- **Statistics of chain-coded stroke contours**
  \[ \theta = \tan^{-1}\left( \frac{n_1 + n_2 + n_3}{n_1 - n_3} \right) \]
Conventional techniques : the problem

Assumption in all conventional techniques :

*Slant is uniform in a word*

But…. sometimes, slant is **non-uniform**
(with a probability of 5 to10%)
Overview of the present technique

- Estimation and correction of non-uniform slant in an optimization framework

- Simple and fast algorithm based on dynamic programming (DP)
Non-uniform slant correction as an optimal estimation problem

input image

local slant angle $\theta_i$

position

non-uniform slant correction = optimal estimation of $\theta_1, \ldots, \theta_i, \ldots, \theta_M$
Create slant-corrected image using estimated slant

slant estimation

map each slant line to be vertical

output

day

day
Directions on designing the criterion for the optimal estimation

Detect long vertical strokes

and

Propagate their slant angles to their neighborhood smoothly
Our criterion

\[
\text{maximize} \quad \sum_{i=1}^{M} [s_i(\theta_i) + \rho(\theta_i | \theta_{i-1})]
\]

- \(s_i(\theta_i)\): confidence level of \(\theta_i\) at position \(i\)
- \(\rho(\theta_i | \theta_{i-1})\): continuity between \(\theta_i\) and \(\theta_{i-1}\)
Algorithm based on dynamic programming

\[
g_i(\theta_i) = s_i(\theta_i) + \max_{\theta_{i-1}} \left[ g_{i-1}(\theta_{i-1}) + \rho(\theta_i \mid \theta_{i-1}) \right]
\]
Computational complexity

- **Theoretical**: \( O(MNW) \)
  where \( M = \) width, \( N = \) height, and \( W = \) max slant

- **Practical**: 140 ms (latest result)
  for \( M = 256, \; N = 64, \; \text{and} \; W = 60 \text{(degrees)} \)
  at PC with Pentium III, 500MHz
Results (1) : Correction of near uniform slant

original (CEDAR CDROM)

conventional technique

present technique

<table>
<thead>
<tr>
<th>Original</th>
<th>Conventional</th>
<th>Present</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buffalo</td>
<td>Buffalo,</td>
<td>Edmond</td>
</tr>
<tr>
<td></td>
<td>Buffalo,</td>
<td>Edmond</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Edmond</td>
</tr>
</tbody>
</table>
Results (2) : Correction of non-uniform slant

original (CEDAR CDROM)

conventional technique

present technique

\[
\begin{array}{c}
\text{original} \\
\text{conventional} \\
\text{present}
\end{array}
\begin{array}{c}
\text{technique} \\
\text{technique}
\end{array}
\]
Quantitative evaluation (1)

- Mean estimation error was measured using artificially slanted words

- Two slant patterns were used
  - Constant slant
  - Sinusoidal slant

- 5 city name words were subjected
  - (Kentucky, California, Memphis, New York, Texas)
Quantitative evaluation (2)

**constant slant**

- **Kentucky**
  - Present technique
  - Conventional technique

**sinusoidally changing slant**

- **Kentucky**

---

Human Interface Lab.  Kyushu Univ.
Remaining problem: over-correction

- **Examples**

  ![Examples]

  **original**
  - Saginaw
  - My
  - Texas

  **present technique**
  - Saginaw
  - My
  - Texas

- **Cause:** originally slanted parts
Conclusion of first part

- Formulate the non-uniform slant correction problem as an optimization problem
- Provide a simple and fast algorithm based on dynamic programming
- Indicate superiority over conventional techniques through experiments
Topic 2 (planning phase)

Non-uniform skew correction for scanned document
Purpose and methodology

- Remove non-uniform skew in scanned document
- Use an algorithm similar to the foregoing non-uniform slant correction algorithm
Types of non-uniform skew (1): Skew due to paper feeding

(High-speed) Sheet-fed scanner

slip and sticking

non-uniform skew correction by affine transformation
Types of non-uniform skew (2): Skew due to the thickness of book

Curvilinear distortion

Flatbed scanner

book
Model of non-uniform skew

non-uniform skew represented as the sequence of $(\theta_i^R, \theta_i^L, j_i)$
Algorithm based on dynamic programming

\[ (\theta_i^R, \theta_i^L, j_i) \]
Future work for non-uniform skew correction

- Collect skewed documents and observe their characteristics
- Design criterion
- Evaluate the technique