6. Applications

- Text recognition in videos -

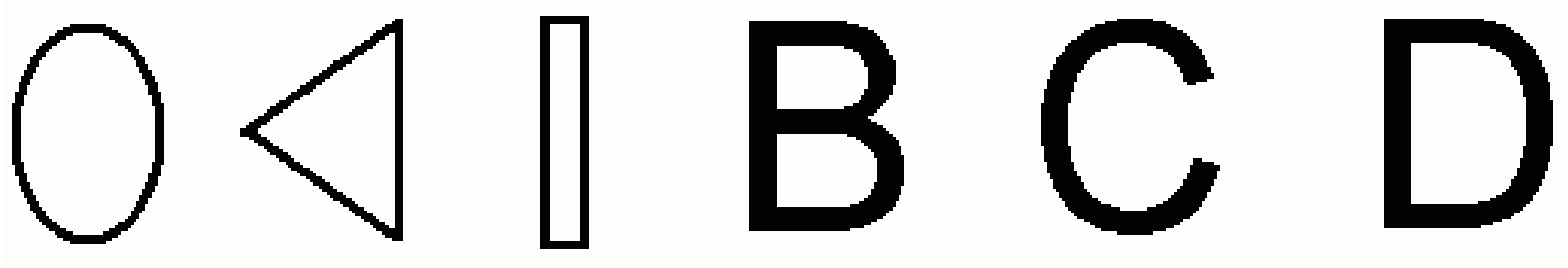
Semantic video analysis

Stephan Kopf

Motivation

Goal:

- Segmentation and classification of characters



- Only **few significant features** are visible in these simple shapes.
- Is it possible to use the curvature scale space approach for classification?

Overview

Optical character recognition (OCR)

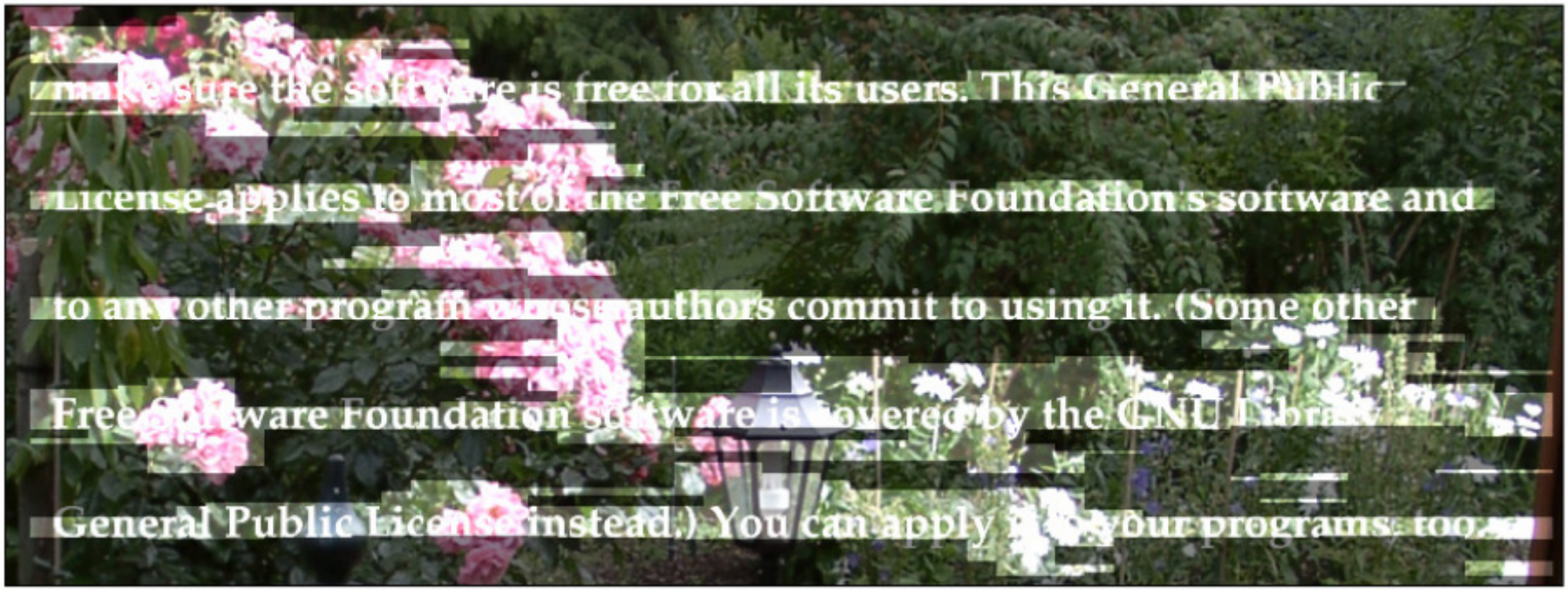
1. Identification of text regions
2. Segmentation of characters
3. Selection of text pixels
4. Character recognition
   - Pattern matching
   - Zoning
   - Shape contexts
   - Contour profiles
   - Curvature scale space

Identification of text regions (I)

Approach

1. Identify blocks with strong edges
2. Aggregate blocks to text regions
3. Use horizontal projection profiles to identify text lines

Block identification



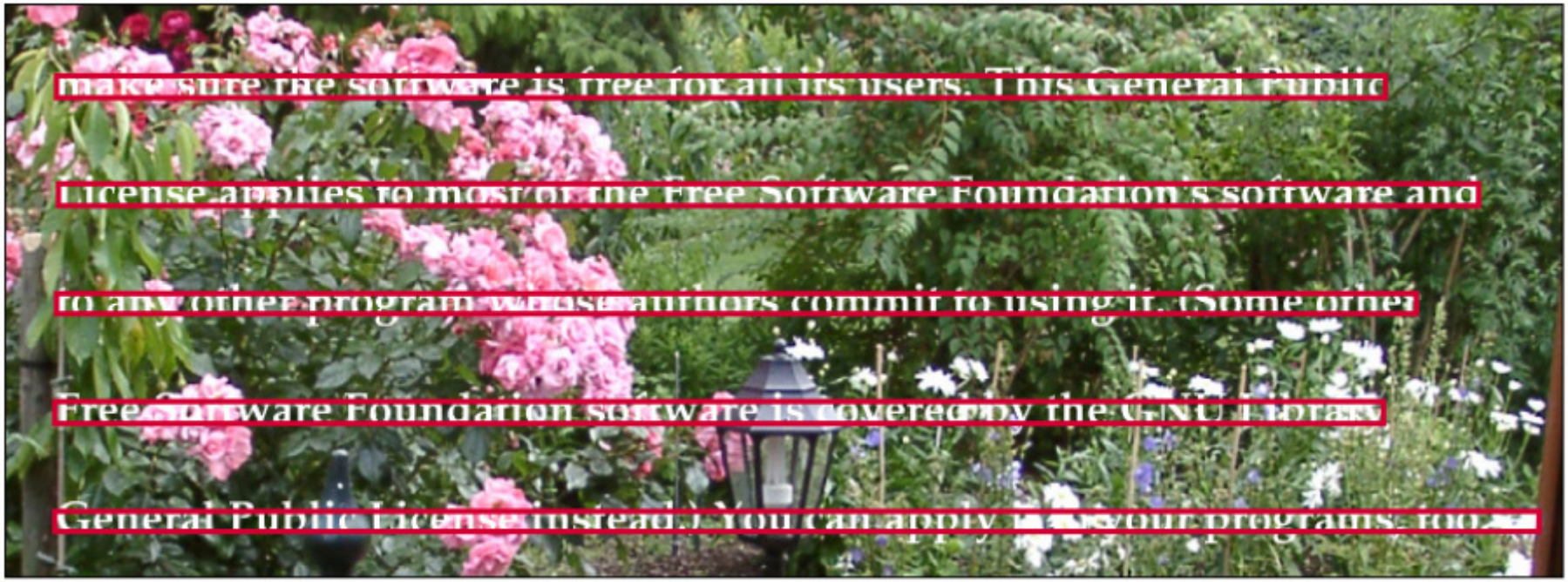
Identification of text regions (II)

Aggregation of blocks

Identification of text regions (III)

Horizontal projection profiles

Identified text lines



Segmentation of characters (I)

First approach

Use vertical projection profiles to identify separators between characters.

Problem

- Split or merged characters

Segmentation of characters (II)

New idea

• Find **cheapest path** as separator for characters.

- Search a path from top to bottom of each text line.
- Minimize the costs of the path.
- Costs are defined as the sum of absolute differences of two consecutive path pixels.
- Use Dijkstra shortest path algorithm to identify the cheapest path.

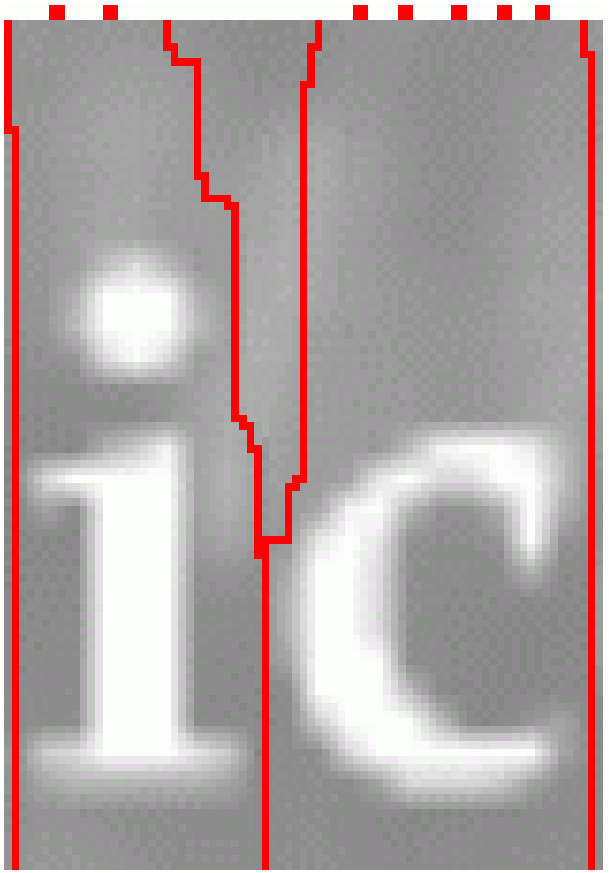
Optimization of the cheapest path algorithm

Problem

- The effort to calculate the cheapest path for each pixel is very high.

Optimization

1. Initialize candidate start pixel.
2. Calculate cheapest path for left and right pixel.
3. Remove candidate start pixels, if cheapest paths merge.
4. Select next candidate start pixel.
5. Continue with 2.



Selection of text pixels (I)

Apply a modified **region merging algorithm** to identify text pixel

- Calculate the histogram of a text region and identify one or two dominant colors. One of these colors defines the text color.
- Apply a region growing algorithm to identify regions in a text line
- Each region in a text line has one of three states: text, background or undefined. All regions are undefined first.
- Set all regions with text color to text regions.
- Undefined regions at the top or bottom of a text line are set as background regions.

Selection of text pixels (II)

6. Calculate distance $\boldsymbol{D_{i,j}}$ between an undefined region ***i*** and defined regions ***j*** (text or background) based on colors $\boldsymbol{C_i}$ and center of gravity $\boldsymbol{G_i}$:

$$D_{i,j} = |C_i - C_j| + |G_i - G_j|.$$

7. Select minimum distance $\boldsymbol{D_{i,j}}$ and define region as text or background region.
8. Continue with step 6 until all undefined regions are defined.

Character recognition (I)

Compare 5 techniques for OCR

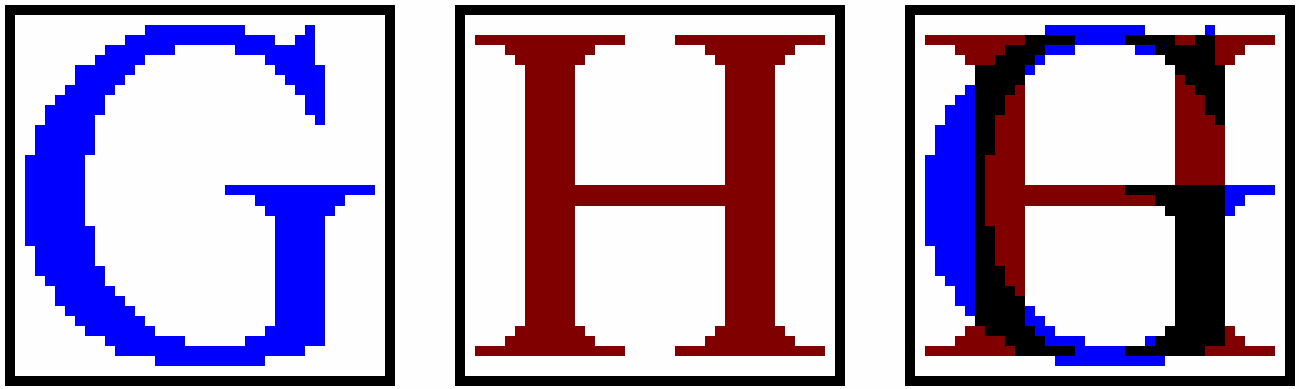
- Pattern matching
- Zoning
- Shape contexts
- Contour profiles
- Curvature scale space

Character recognition (II)

Pattern matching

Calculate pixel difference between two binary images:

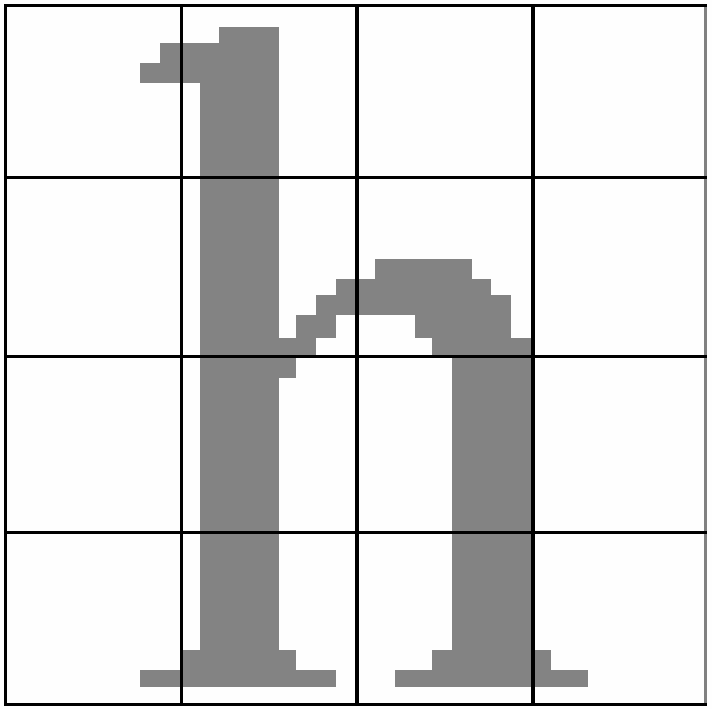
$$D_{Q,J} = \frac{1}{n_x \cdot n_y} \cdot \sum_{x=1}^{n_x} \sum_{y=1}^{n_y} \begin{cases} 0 & \text{falls } Q_{x,y} = J_{x,y}, \\ 1 & \text{sonst.} \end{cases}$$



Character recognition (III)

Zoning

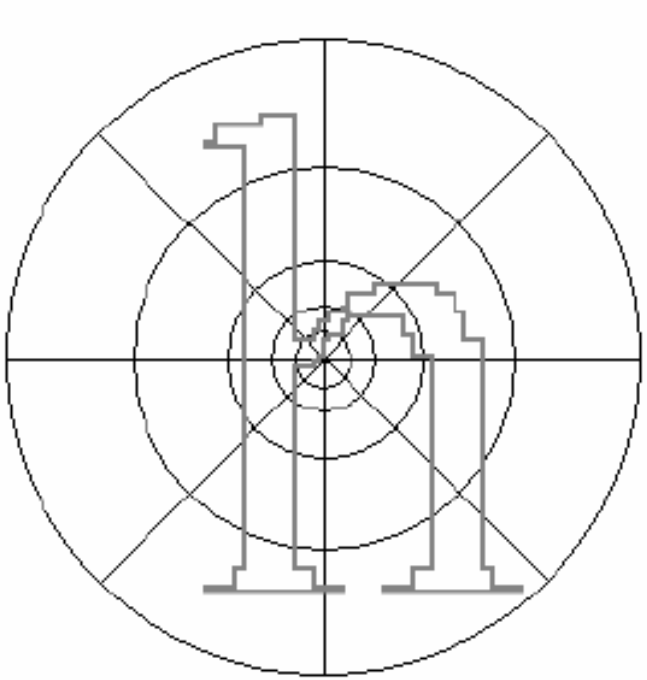
- Define a grid of N x M cells.
- Count number of text pixels in each cell.
- Compare two vector which are defined by the number of pixels in each cell.



Character recognition (IV)

Shape contexts

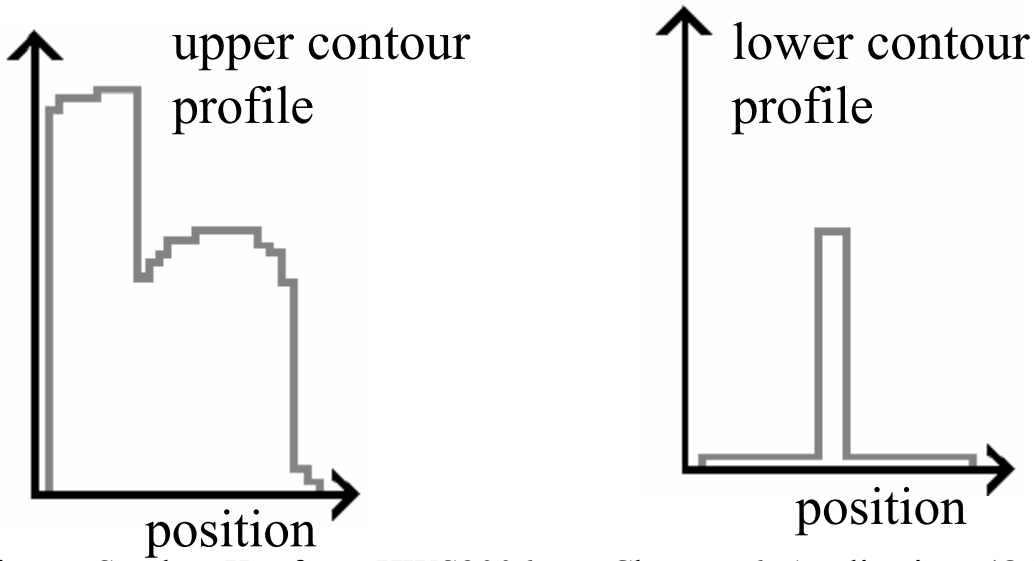
- The shape context algorithm is a specialized zoning algorithm.
- Circular segments are used to define the cells.
- One shape pixel defines the center of the circle.
- The number of shape pixel in each cell defines the feature vector.
- Different shape pixels should be selected as circle centers to get reliable matching results.



Character recognition (V)

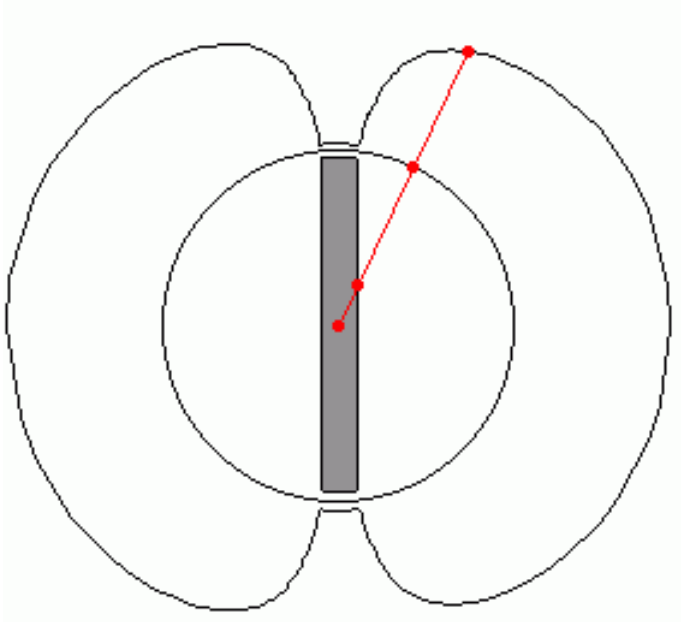
Contour profiles

- Horizontal profile: analyze contour pixels at the top and at the bottom.
- Vertical profile: analyze left and right contour pixels.
- Aggregate these four contour profiles in one vector.



Character recognition (VI)

Curvature scale space approach with mapped shapes



Approach:

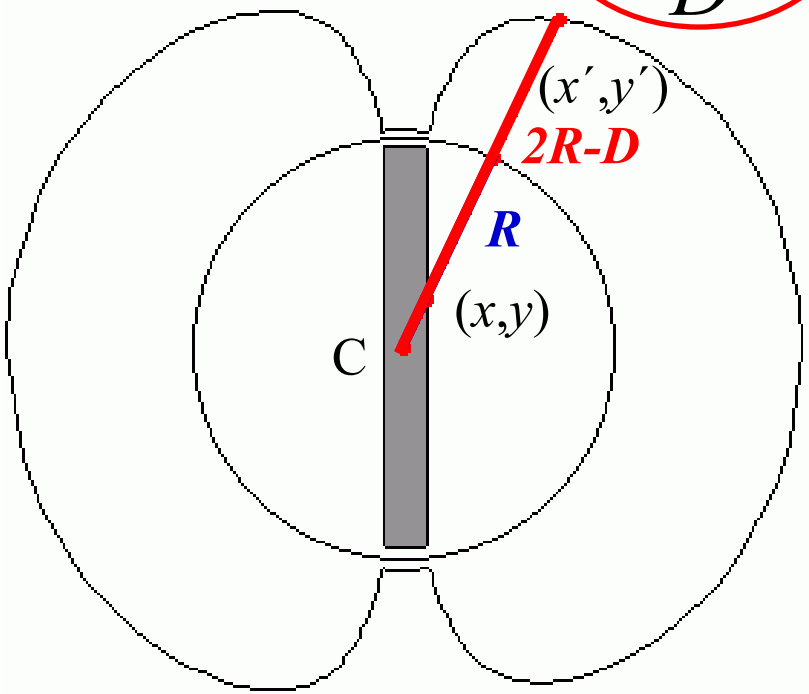
- Identify center of gravity.
- Put a circle around the shape.
- Reflect each shape pixel on this circle.

→ Strong convex segments of the original shape become concave segments of the mapped shape.

Character recognition (VII)

Curvature scale space approach with mapped shapes

$$x' = \frac{2R - D}{D}(x - C_x) + C_x$$

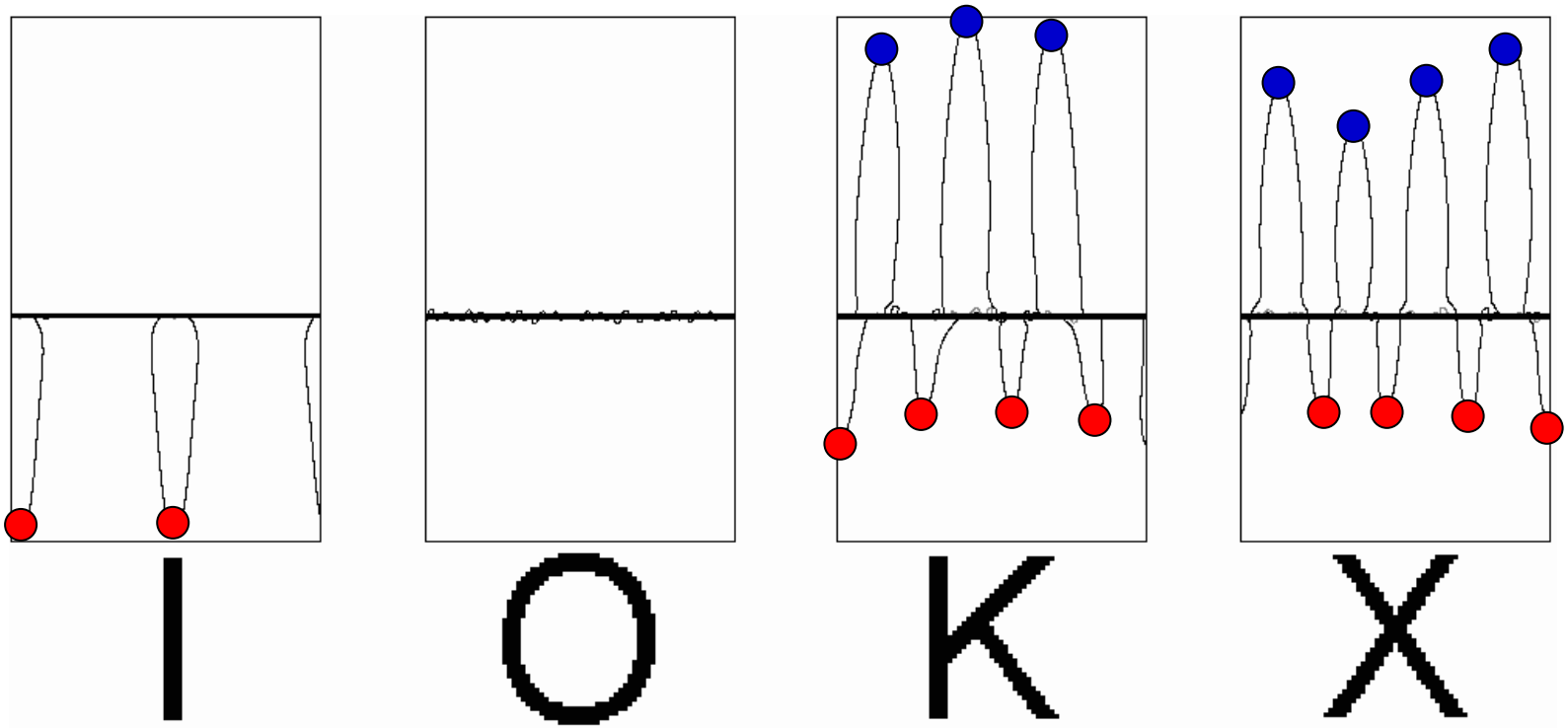


(x,y)	shape pixel
(x',y')	mapped shape pixel
$C=(C_x,C_y)$	center of gravity
R	radius of circle
D	distance between shape pixel and C
$2R - D$	distance between (x',y') and C

Character recognition (VIII)

Curvature scale space approach with mapped shapes

- Get standard curvature scale space features.
- Calculate features for the mapped shape.



Experimental Results (I)

Database

- Characters of 4 fonts are stored.
- Limit rotation of characters to ~20 degrees to recognize italic characters.

Challenges

- OCR with segmentation errors:

Experimental Results (II)

Segmentation errors	Projection profile	Cheapest path
Characters split	9.9 %	3.8 %
Characters merged	7.5 %	5.4 %
Segmentation errors	17.4 %	9.2 %

OCR technique	Recognition Results
Pattern Matching	69 %
Zoning	64 %
Contour profiles	71 %
Curvature scale space approach	76 %
Commercial OCR	75 %

Experimental Results (III)

Questions ?