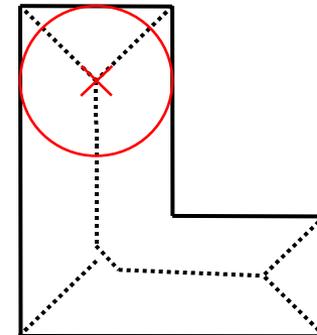
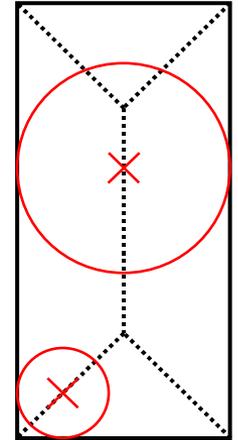


# Erkennung einzelner Buchstaben (VI)

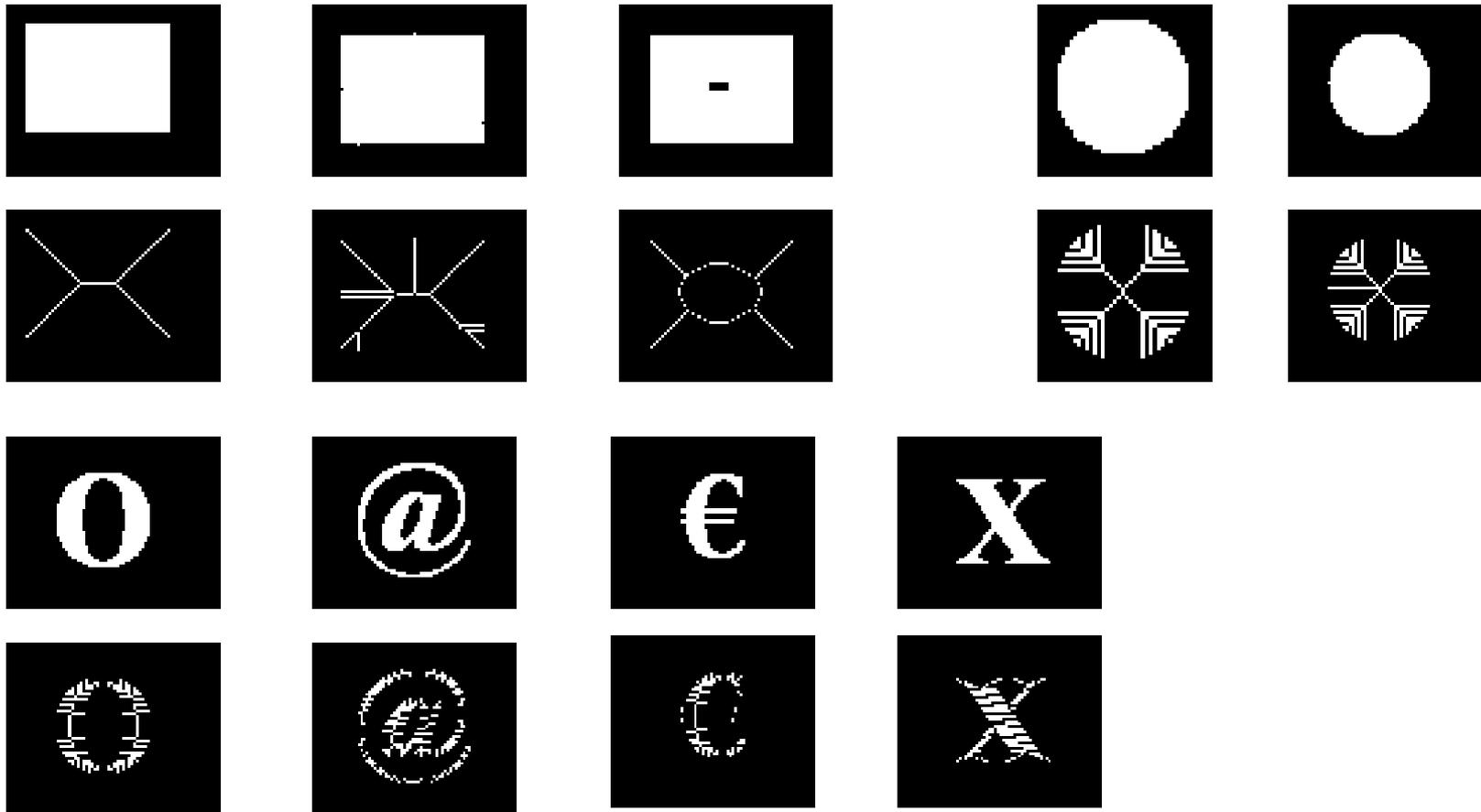
## Skelette

- Idee: Ermittle die Struktur einer Region
- Vorgehen: Trage die Region iterativ ab
- Im Jahr 1967 wurde das Verfahren der *Medial-Axis-Transformation* vorgestellt:
  - R: Region, B: Rand
  - Für alle  $p \in R$ : Suche nächsten Nachbarn (z.B. mittels City-Block-Distanz) in B.
  - Falls (Anzahl Nachbarn  $> 1$ ): Pixel ist mediale Achse (Skelett) von R



# Erkennung einzelner Buchstaben (VII)

Beispiel für Skelette bei Verwendung der City-Block-Distanz



# Erkennung einzelner Buchstaben (VIII)

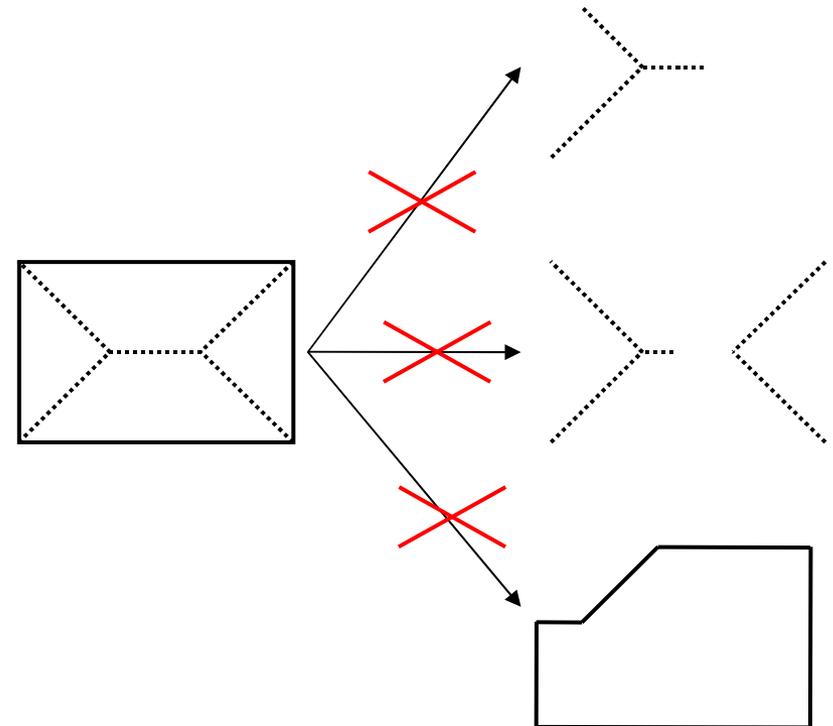
## Probleme bei der Erzeugung von Skeletten

- Minimale Änderungen der Kontur ergeben sehr unterschiedliche Skelette.
- Der Rechenaufwand ist sehr hoch, da für jedes Pixel die Distanz zu jedem anderen Pixel berechnet werden muss.
- **Optimierung:**  
Trage Regionen ab (entferne Randpixel), so dass:
  - Endpunkte des Skeletts möglichst wenig abgetragen werden,
  - eine Region nicht in zwei Regionen unterteilt wird,
  - alle Regionen des Objektes gleichmäßig stark abgetragen werden (eine Region soll nicht übertrieben stark abgetragen werden).

# Erkennung einzelner Buchstaben (IX)

## Bedingungen

- Endpunkte des Skeletts sollen möglichst wenig abgetragen werden,
- eine Region soll nicht in zwei Regionen unterteilt werden,
- alle Regionen des Objektes sollen gleichmäßig stark abgetragen werden (eine Region soll nicht übertrieben stark abgetragen werden).



# Erkennung einzelner Buchstaben (X)

## *Thinning-Algorithmus zur Erzeugung von Skeletten*

- Gegeben: Binärbild (Hintergrund=0, Objekt=1)

- 8-Pixel-Nachbarschaft

aktuelles Pixel:  $p_1$

$p_9$	$p_2$	$p_3$
$p_8$	$p_1$	$p_4$
$p_7$	$p_6$	$p_5$

4. Betrachte jedes Randpixel und markiere Pixel falls alle Bedingungen erfüllt sind:

- $3 \leq N(p_1) \leq 6$
- $S(p_1) = 1$
- $p_2 * p_4 * p_6 = 0$
- $p_4 * p_6 * p_8 = 0$

5. Lösche markierte Pixel

# Erkennung einzelner Buchstaben (XI)

## *Thinning-Algorithmus zur Erzeugung von Skeletten*

2. Betrachte jedes Randpixel und markiere Pixel falls alle Bedingungen erfüllt sind:

- $3 \leq N(p_1) \leq 6$
- $S(p_1) = 1$
- $p_2 * p_4 * p_8 = 0$
- $p_2 * p_6 * p_8 = 0$

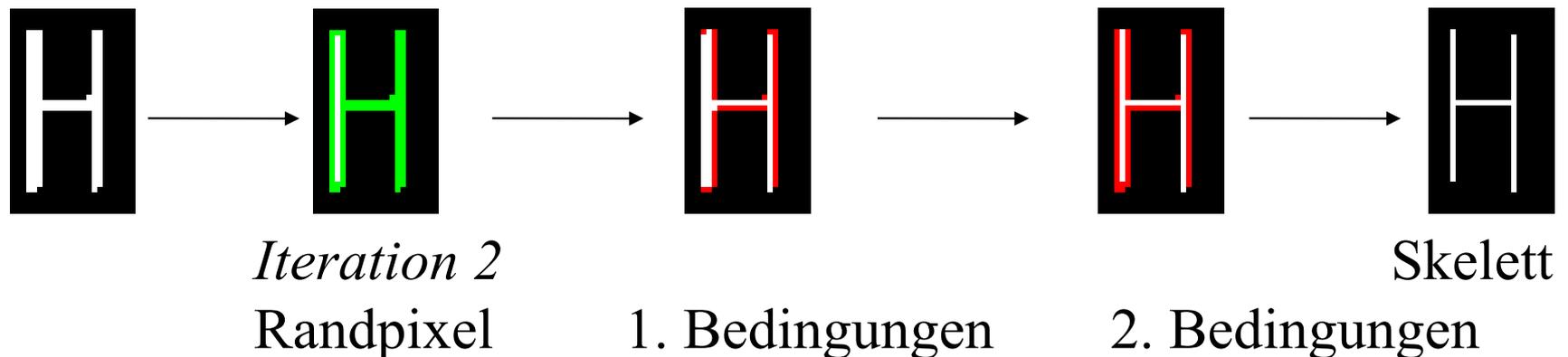
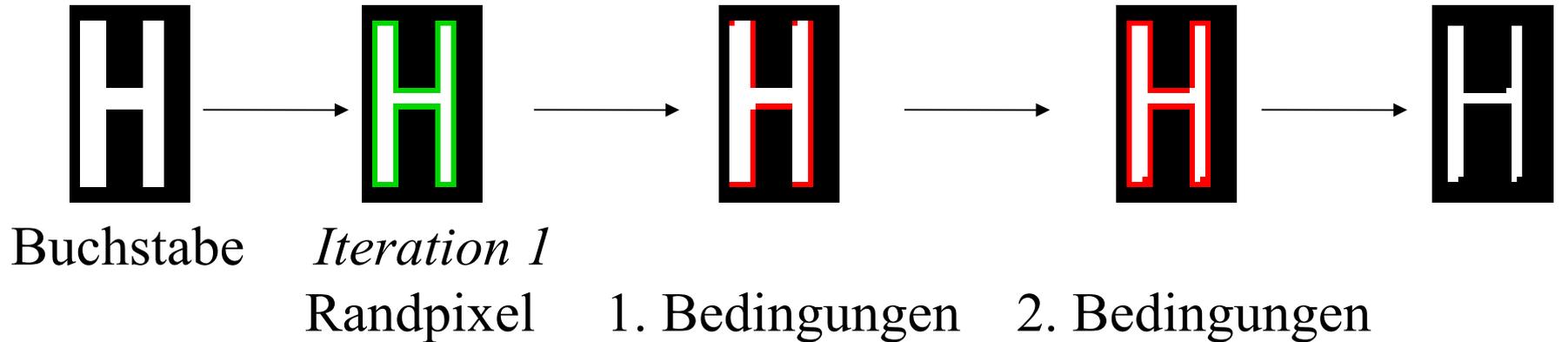
$p_9$	$p_2$	$p_3$
$p_8$	$p_1$	$p_4$
$p_7$	$p_6$	$p_5$

3. Lösche markierte Pixel

4. Gehe zu 1. falls mindestens ein Pixel gelöscht wurde.

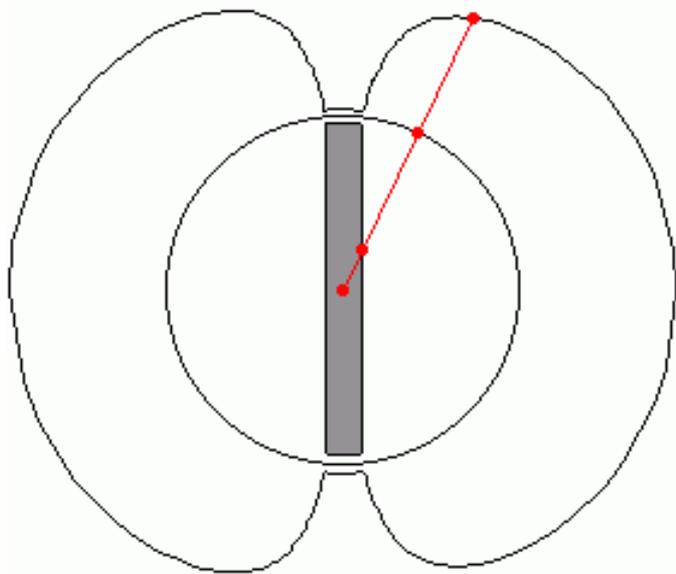
# Erkennung einzelner Buchstaben (XII)

*Thinning-Algorithmus zur Erzeugung von Skeletten*



# Erkennung einzelner Buchstaben (XIII)

## Texterkennung mit Skalenraumabbildungen



### Ablauf:

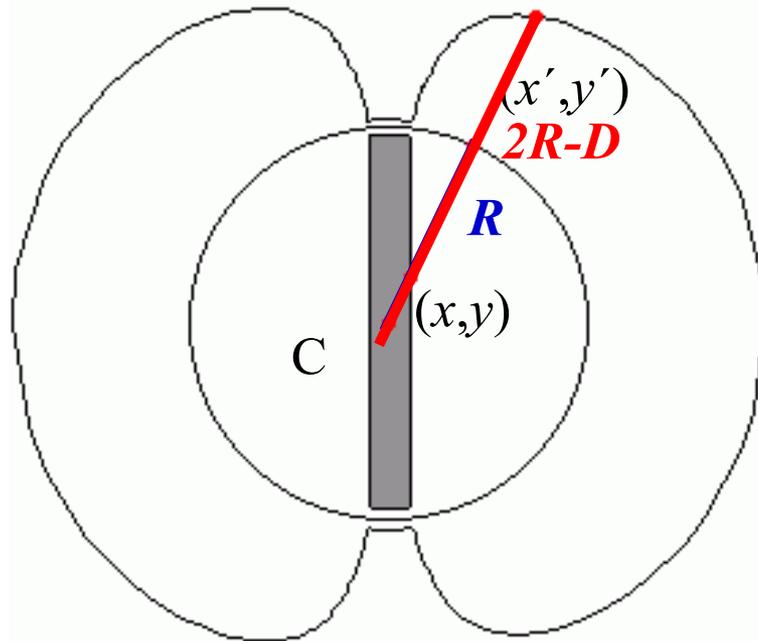
- Identifiziere Schwerpunkt
- Lege Kreis um Buchstaben
- Spiegele Konturpixel des Buchstabens an der Kreislinie

→ Stark konvex gekrümmte Regionen werden zu konkaven Regionen.

# Erkennung einzelner Buchstaben (XIV)

## Texterkennung mit Skalenraumabbildungen

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



$(x, y)$

Konturpixel

$(x', y')$

Gespiegeltes

Konturpixel

$C = (C_x, C_y)$

Schwerpunkt

$R$

Radius des Kreises

$D$

Entfernung zwischen  
Konturpixel und  $C$

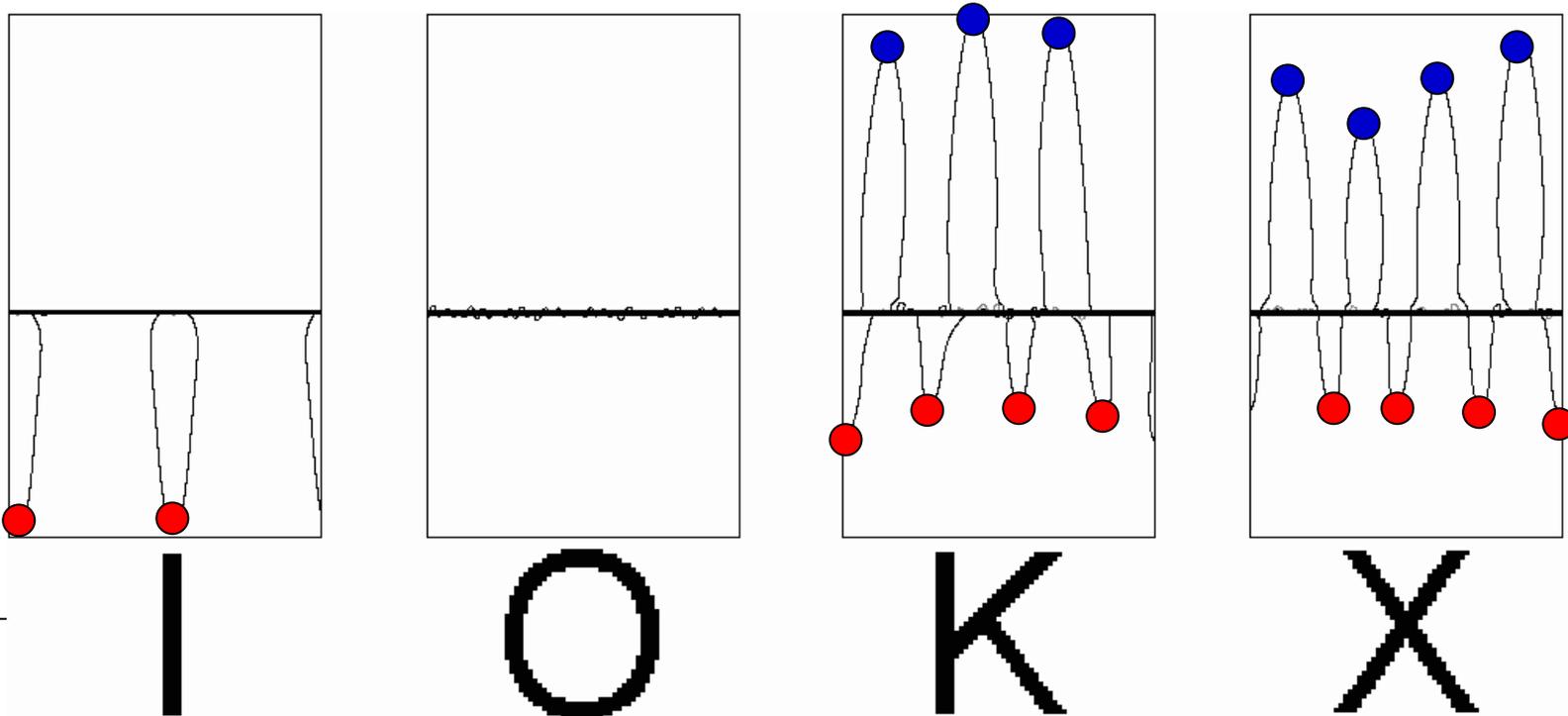
$2R - D$

Entfernung zwischen  
 $(x', y')$  und  $C$

# Erkennung einzelner Buchstaben (XV)

## Texterkennung mit Skalenraumabbildungen

- Berechne normale Skalenraumabbildung
- Berechne Skalenraumabbildung für gespiegelte Kontur



# Experimentelle Ergebnisse (I)

## Datenbank

- Buchstaben von vier Schriftarten wurden verwendet.
- Die Skalenraumabbildungen durften maximal ~20 Grad gedreht werden, um kursive Zeichen zu erkennen.

## Herausforderungen

- Texterkennung bei Segmentierungsfehlern:

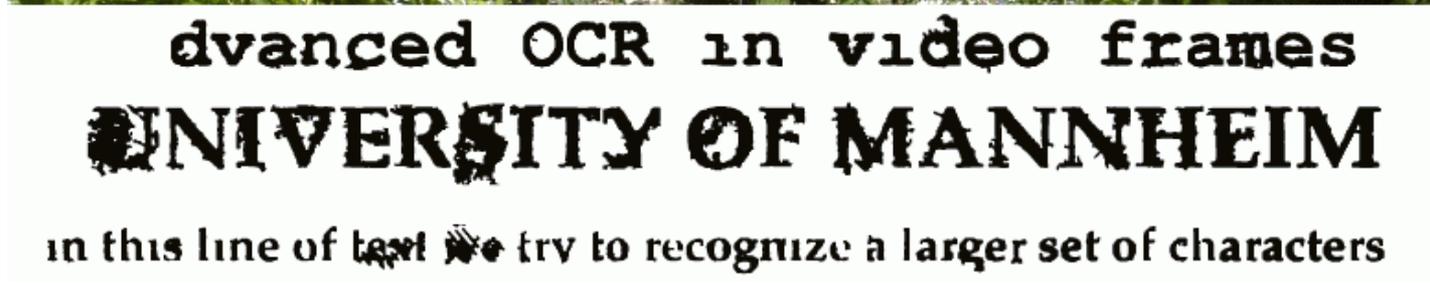
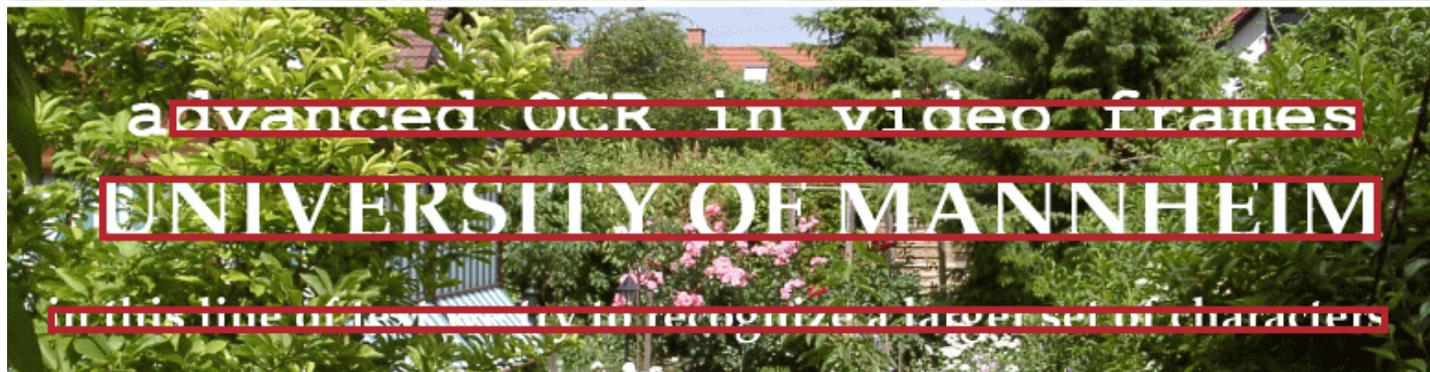
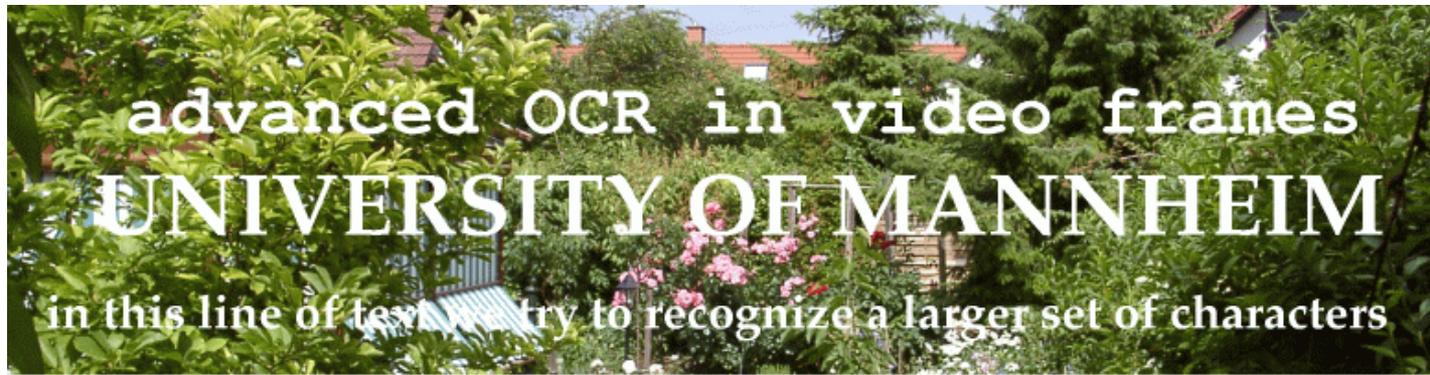


# Experimentelle Ergebnisse (II)

<b>Segmentierungsfehler</b>	<b>Projektionsprofile</b>	<b>Kürzeste-Pfade</b>
Unterteilte Buchstaben	9.9 %	3.8 %
Verbundene Buchstaben	7.5 %	5.4 %
Segmentierungsfehler	17.4 %	9.2 %

<b>Texterkennungsverfahren</b>	<b>Erkennungsergebnisse</b>
Pattern Matching	69 %
Zoning	64 %
Konturprofile	71 %
Skalenraumabbildungen	76 %
Kommerzielle OCR-Software (Scanner)	75 %

# Experimentelle Ergebnisse (III)



# Fragen ?