



## **Wordspotting (segmentierungsfrei) mit einem patchbasierten Ansatz**

**Frederik Heerde, Ben Wilkes, Steven Brodziak, Berat Özdemir, Maximilian Brand**

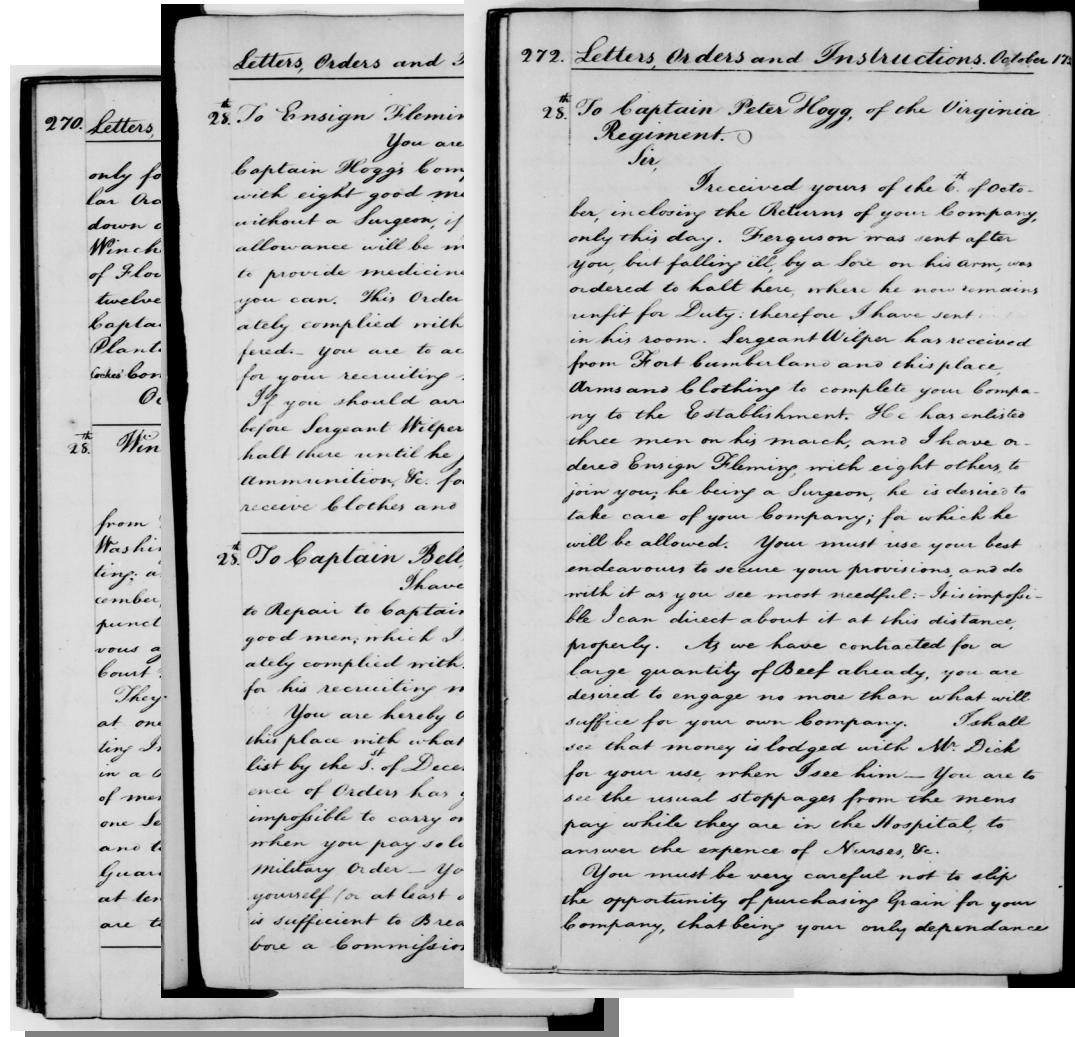
# Unsere Aufgabe

Letters,

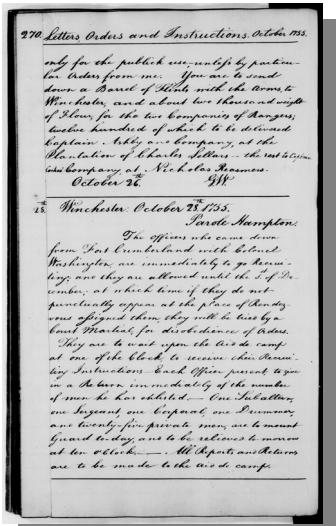
Suchen in:



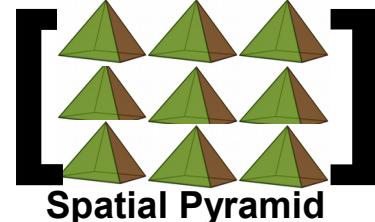
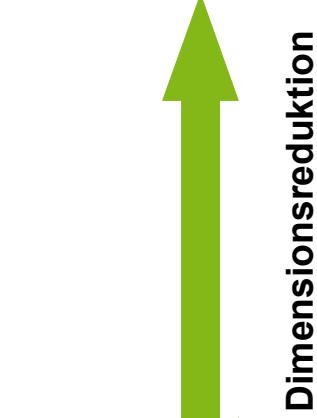
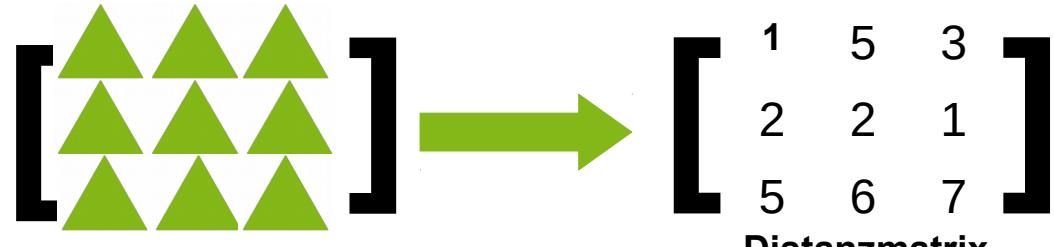
- **Query: Bildausschnitt aus einem Dokument**
- **Ziel: Alle Vorkommen des Querys finden**
- **Patchbasiert**



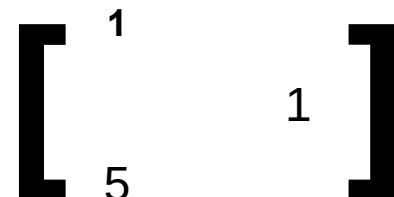
# Unser Vorgehen



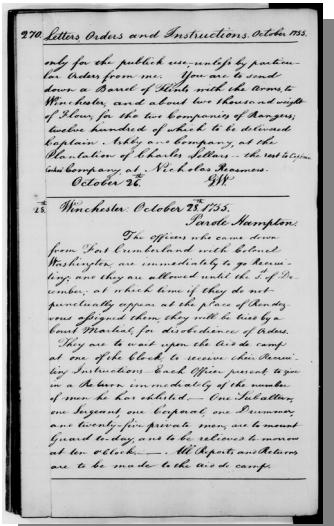
Document Level



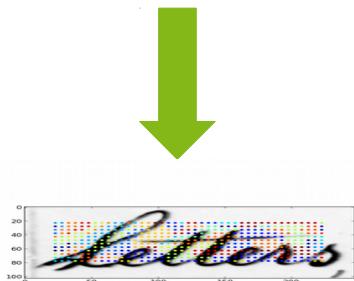
Bag of Visual Words



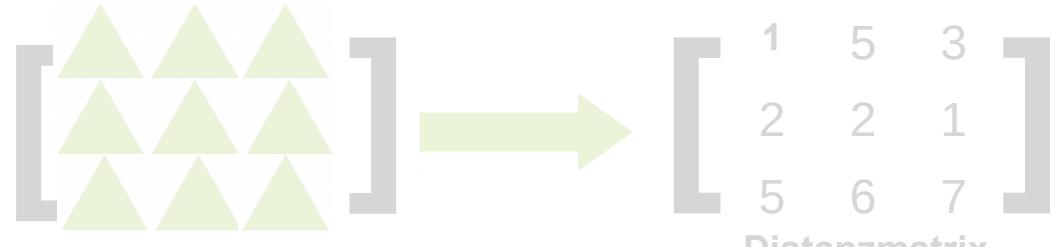
# Unser Vorgehen



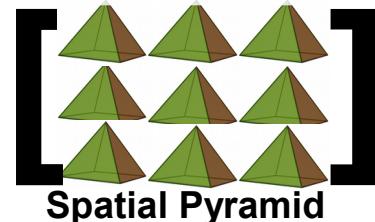
Document Level



Bag of Visual Words

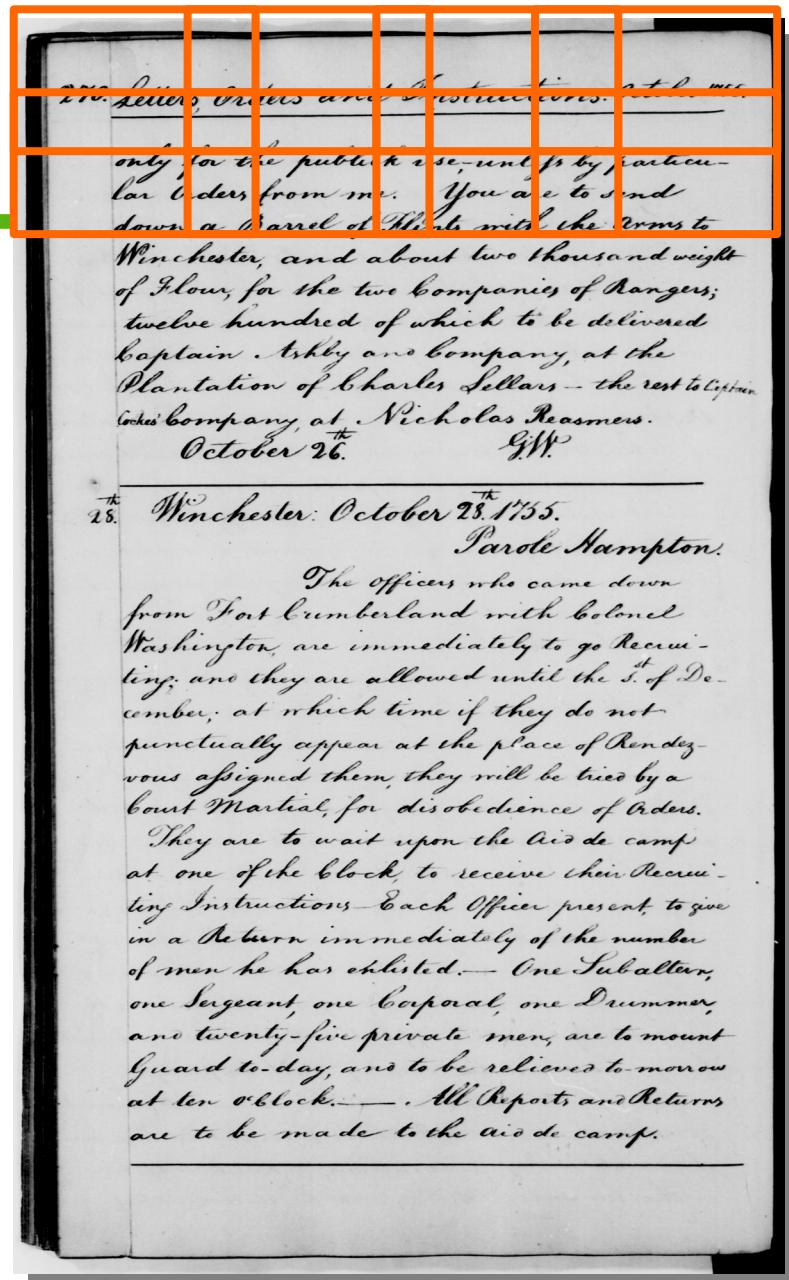


Distanzmatrix



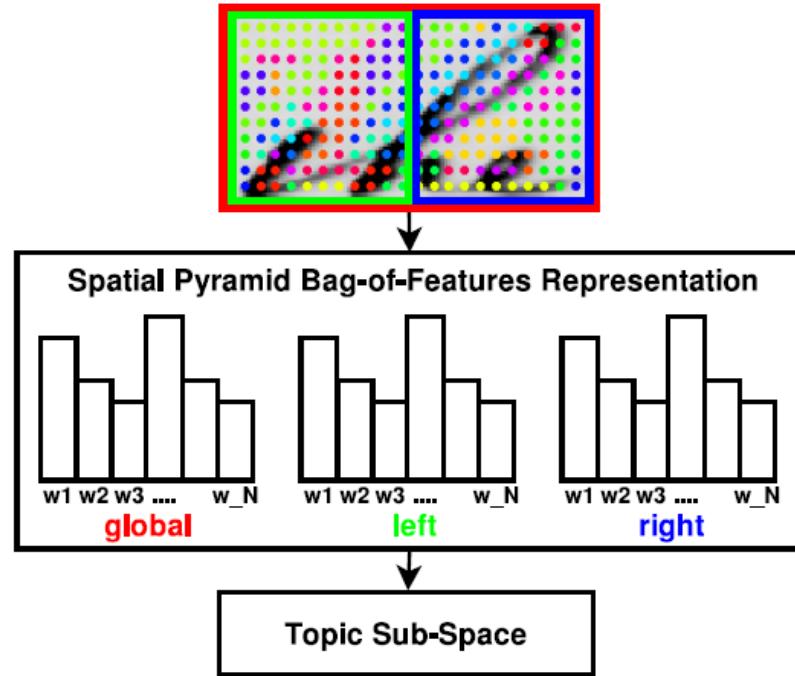
# Patches extrahieren

- Einteilung des Bildes in Bereiche
- Überlappung
- Dynamische Patchbreite nach Querybreite
- Patches beinhalten zu Visual Words quantisierte Sift Deskriptoren



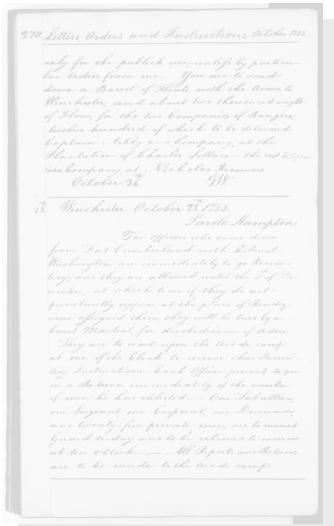
# Spatial Pyramid Berechnung

- Drei Histogramme
  - Gesamter Ausschnitt
  - Linker Ausschnitt
  - Rechter Ausschnitt
- Gesamte Spatial Pyramid aus den Histogrammen konkateniert.
- Histogrammgröße: Größe des Codebooks
- Gesamtgröße:  $3 \times$  Größe des Codebooks

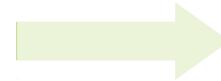
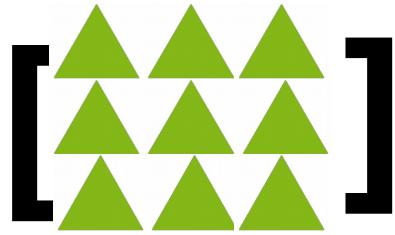


Rothacker, Leonard; Fink, Gernot: „Dokumentenanalyse 2016“, unter: <http://patrec.cs.tu-dortmund.de/lectures/SS16/dokumentenanalyse/document-analysis.pdf> (abgerufen am 05.07.2016)

# Unser Vorgehen



Document Level

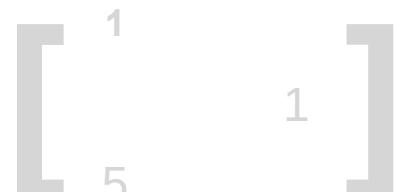
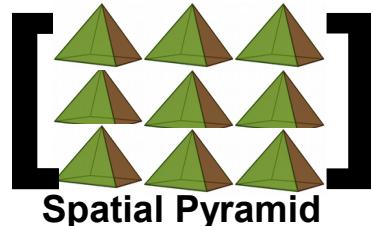


Distanzmatrix

Dimensionsreduktion



Bag of Visual Words



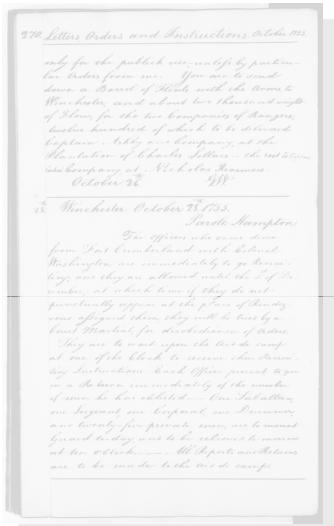
Non-Maximum Suppression

# Dimensionsreduktion

---

- Latent Semantic Indexing
  - Reduziert Anzahl der Dimensionen
  - Erleichtert Distanzberechnung im nächsten Schritt
- 
- Query muss in den gleichen Unterraum transformiert werden

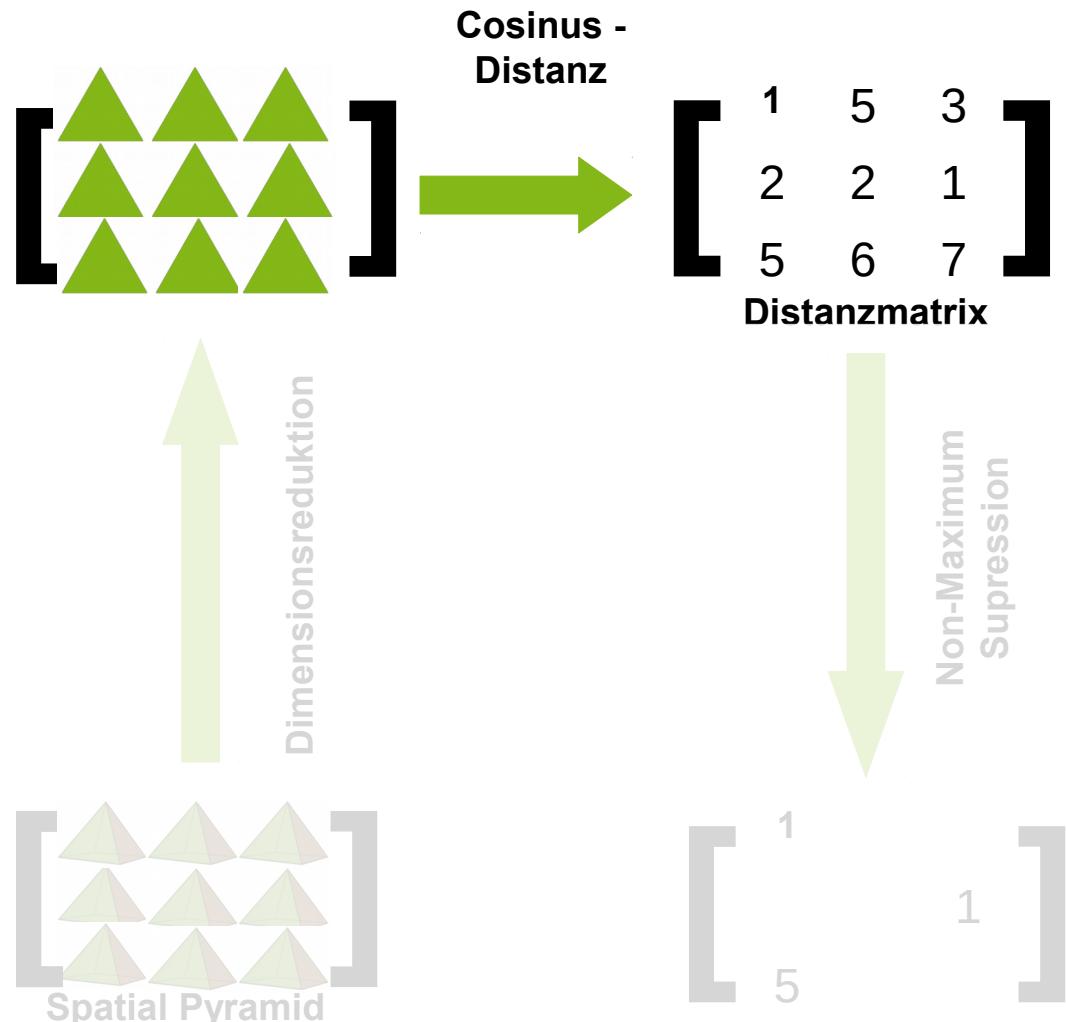
# Unser Vorgehen



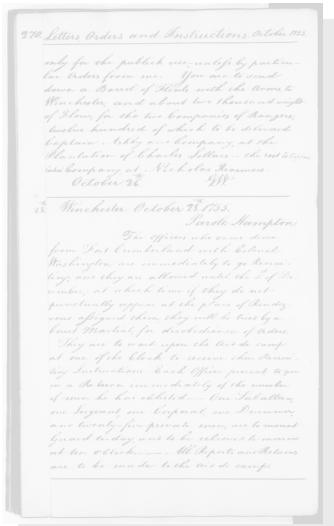
Document Level



Bag of Visual Words



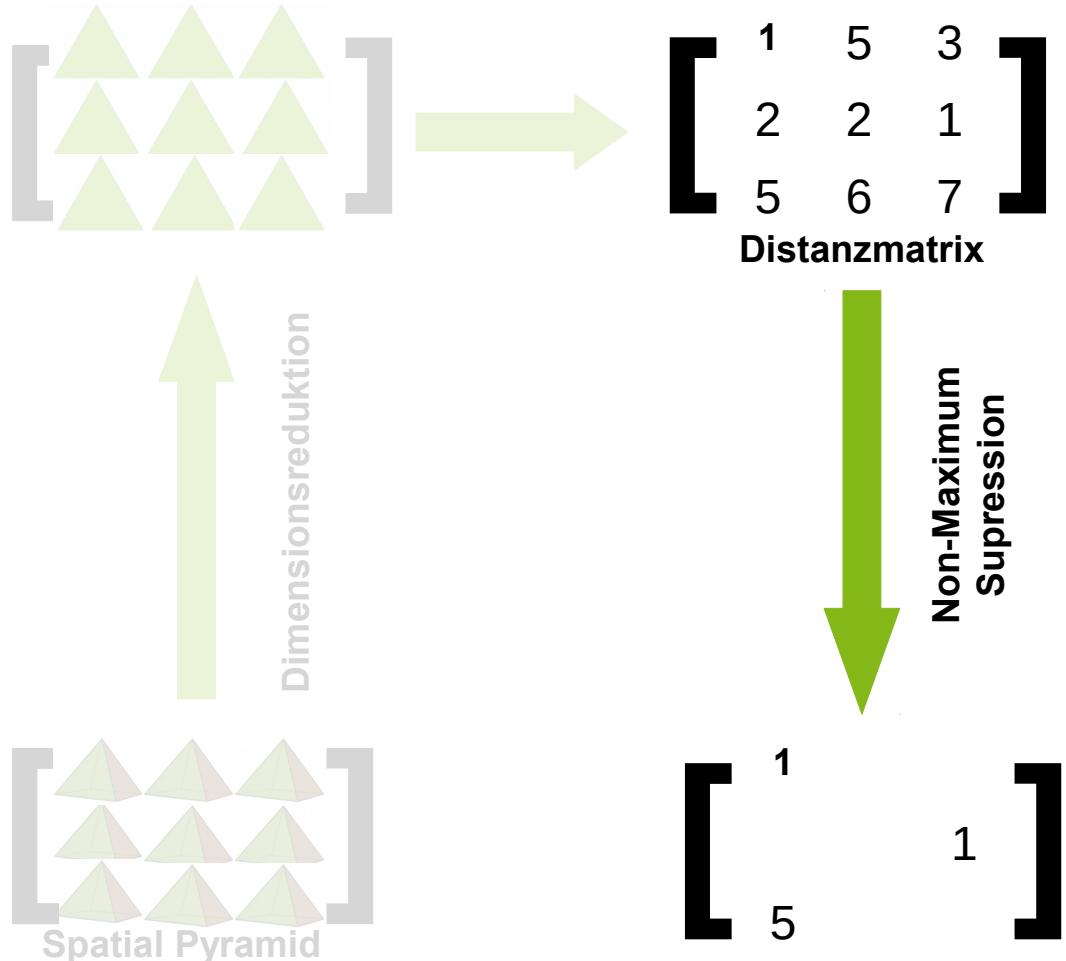
# Unser Vorgehen



Document Level



Bag of Visual Words



# Non-Maximum Suppression

---

- Patches überlappen sich
- Durch Non-Maximum Suppression wird das beste Patch weiter verwendet



- Danach: Patches nach Distanz sortieren und ausgeben

# Quantitative Evaluierung

---

## Konfiguration

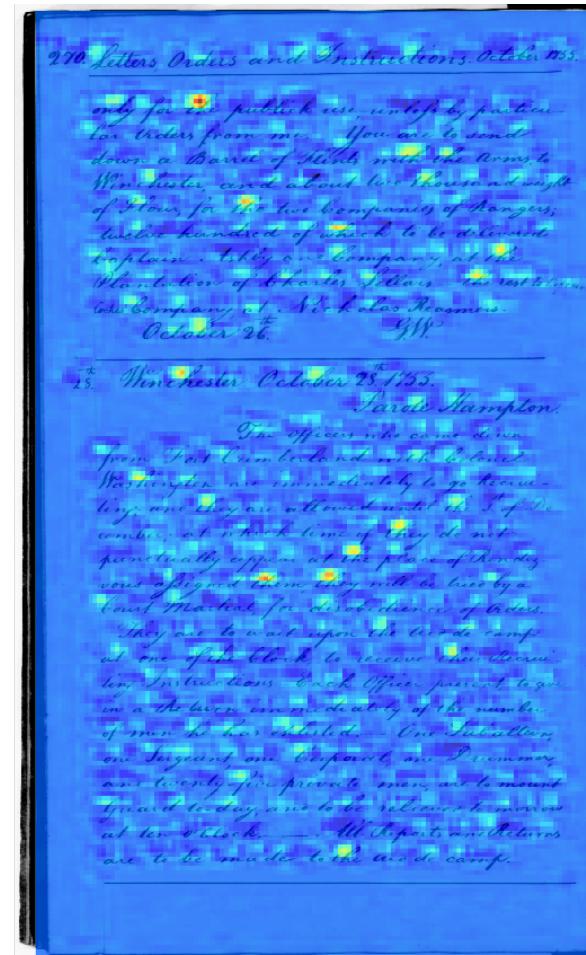
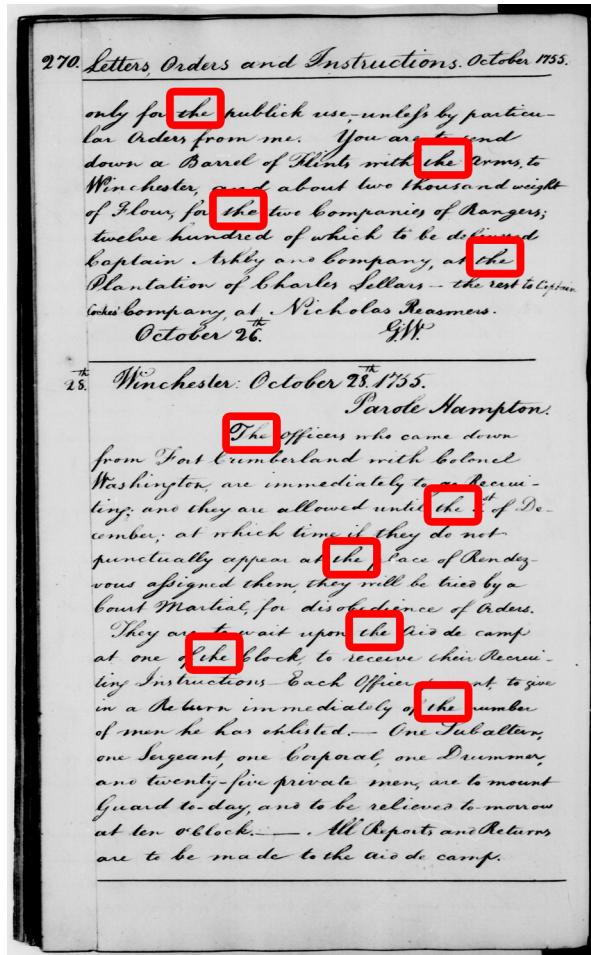
Sift hop_size	5
Sift cell_size	15
Sift n_visual_words	1000
Dimensionen Topic Raum	100
patch_hop_size	20

## Resultate

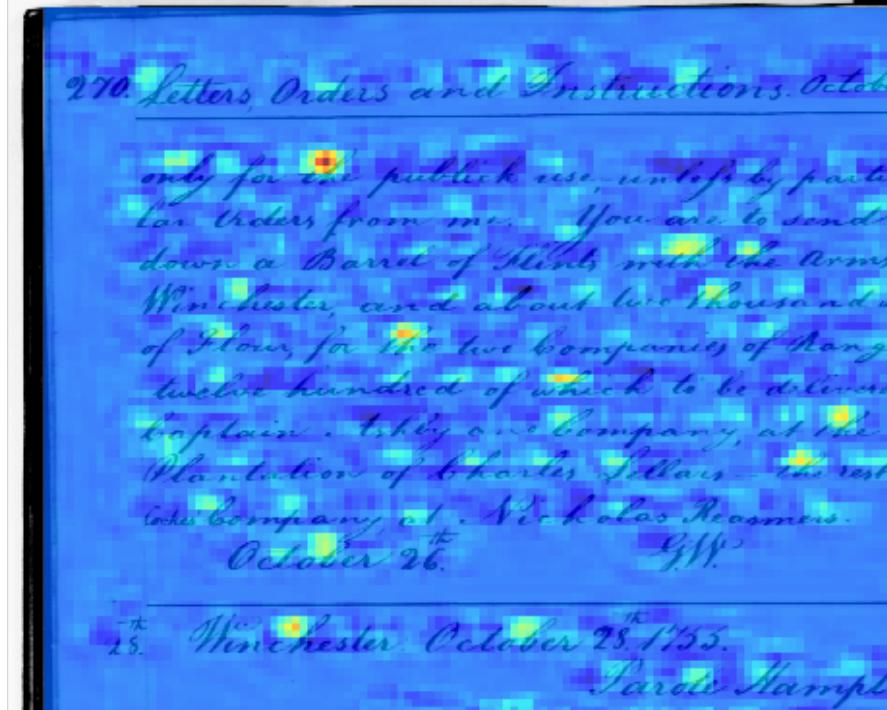
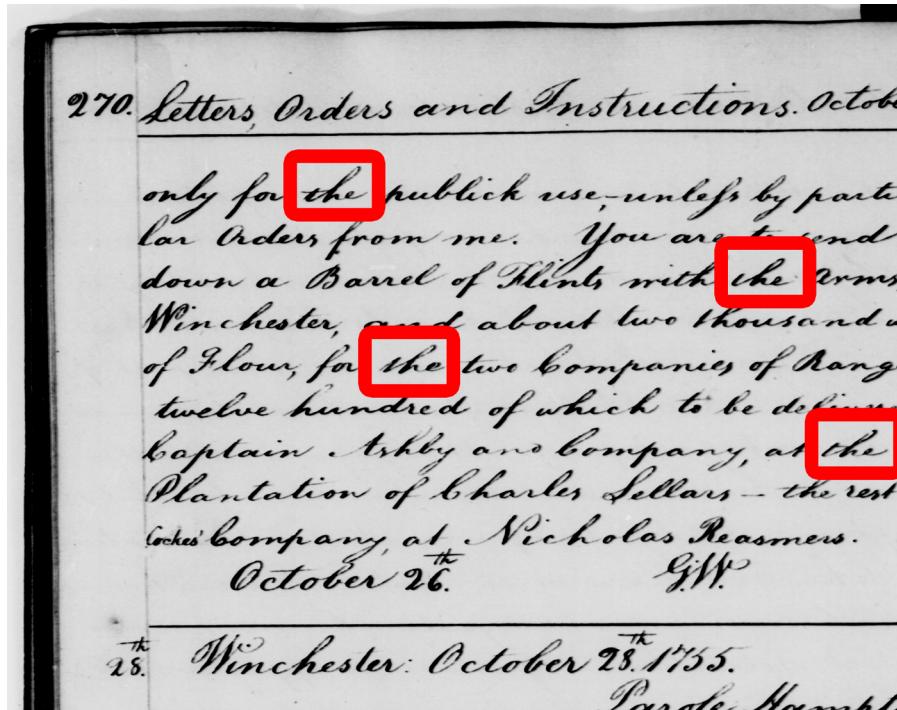
Mean Recall 43,6%

Mean-average-prescion 34%

# Vorkommen von ‚the‘ auf Seite 1



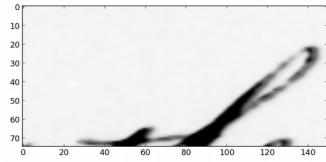
# Vorkommen von ‚the‘ auf Seite 1



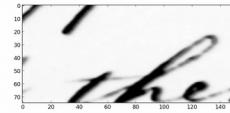
# Ergebnisse

---

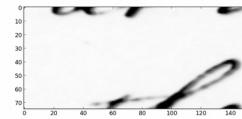
1.



2.



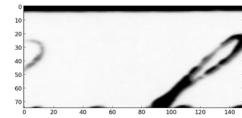
3.



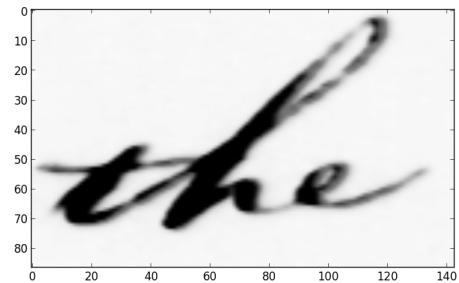
4.



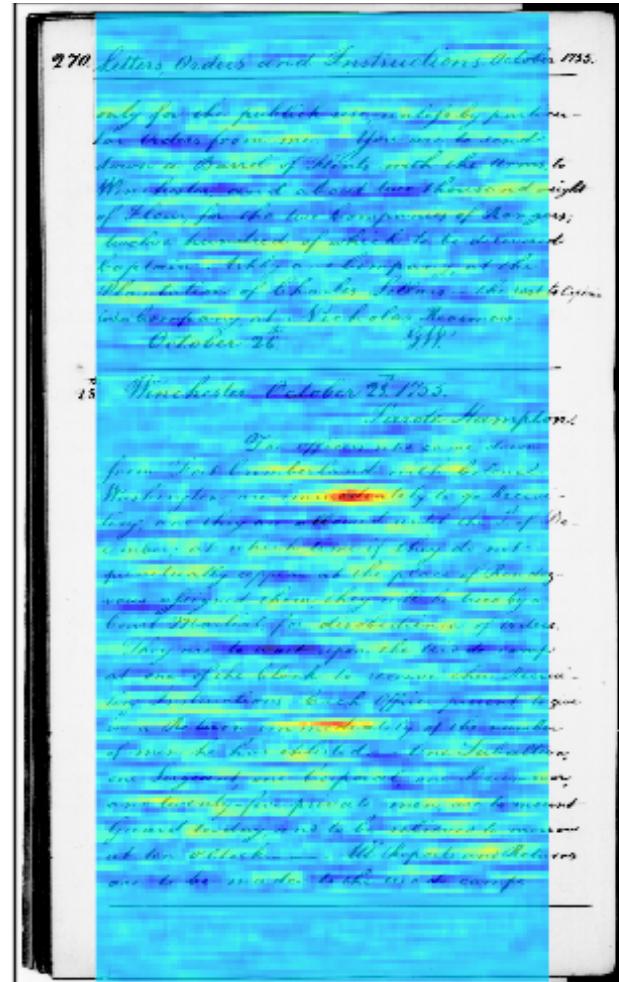
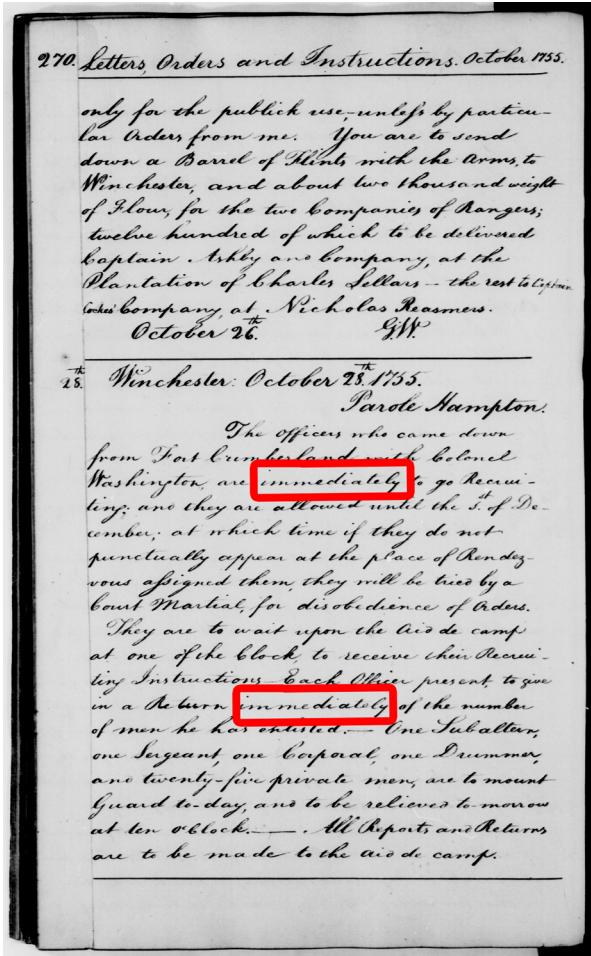
5.



Eingabe Query



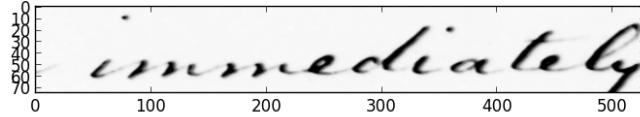
# Vorkommen von „immediatly“ auf Seite 1



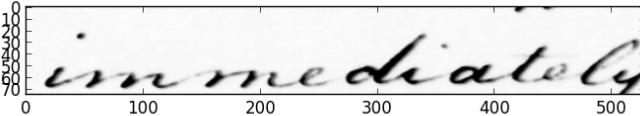
# Ergebnisse

---

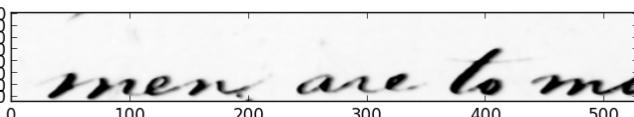
1.



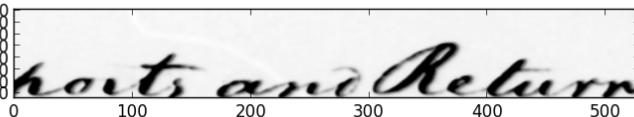
2.



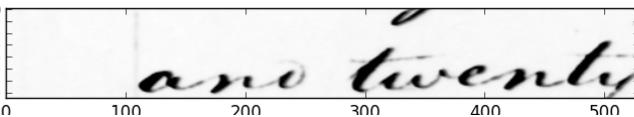
3.



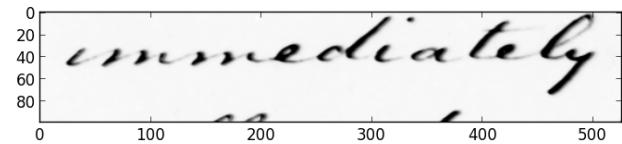
4.



5.



## Eingabe Query



## Fazit

---

- Segmentierungsfreies und patchbasiertes Wordspotting implementiert
- Rückgabe: Liste von Patches, sortiert nach Ähnlichkeit zum Query
- Noch Potential bei der Erkennungsgenauigkeit
- Bestimmte Anfragen geben gute Werte zurück