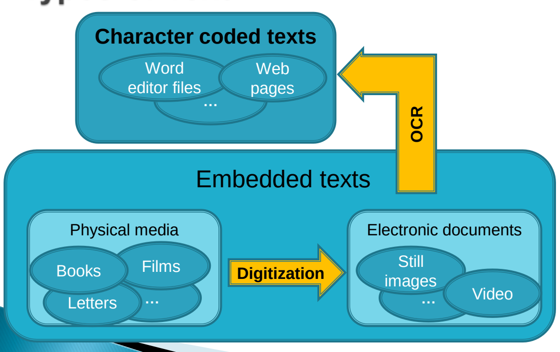
Offline Handwriting Recognition in Archive Documents

Image Processing Laboratory
Department of Electrical Engineering and Information Systems
University of Pannonia

Outline

- OCR (Optical Character Recognition)
- Handwriting recognition
- Document segmentation
- Signature recognition
- Handwriting recognition in archive documents
  - Introduction of the problem
  - Recognition by SIFT points
  - Pivot based search for faster recognition

Types of Text



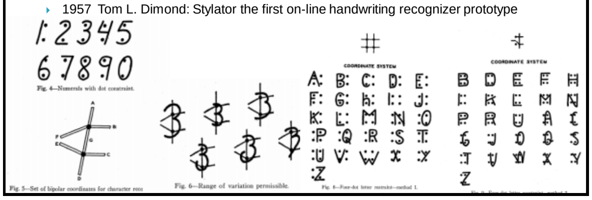
Character (word) recognition

- OCR (Optical Character Recognition)
  - Widespread applications (books, journal papers, etc.)
  - Problems only in noisy/distorted/undersampled environments
- Handwritten text recognition
  - ***Online*** recognition (mobile devices, touchpads, bank signature verification systems), dynamic: uses pen's speed, position, pressure, acceleration, etc.
  - ***Offline*** recognition: uses only static images
- Signature recognition: learn personal characteristics of handwriting (signature verification or writer identification)

History of Handwriting Recognition

- 1914 Hyman Eli Goldberg , U.S. Patent 1,117,184, On-line recognition of hand-written ***numerals*** to control a machine in real-time. Controller: conversion of handwritten numbers to electronic data by inductive ink to controll equipments.

- 1938 George Hansel, U.S. Patent 2,143,875, machine recognition of handwriting
- 1957 Tom L. Dimond: Stylator the first on-line handwriting recognizer prototype



- T. L. DIMOND: Devices for Reading Handwritten Characters

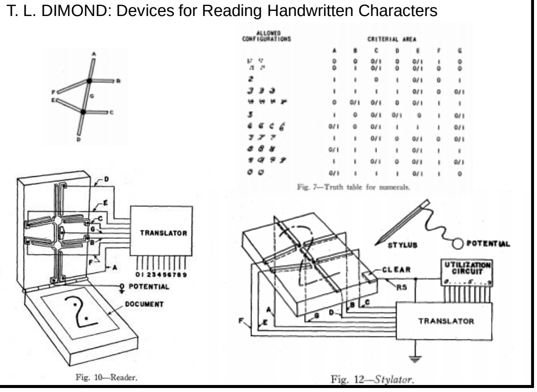
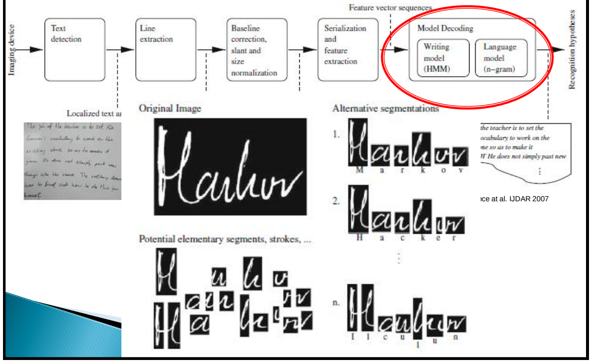


Fig. 7—Truth table for numerals.
Fig. 10—Reader.
Fig. 12—Stylator.

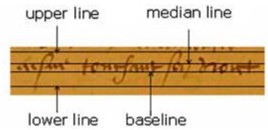
Overview of HWR



Document Sementation for Text Recognition

Aim: to find the correspondence between document image and its content by text/image alignment techniques

- *Baseline: fictitious line which follows and joins the lower* part of the character bodies in a text line (Fig. 1).
- *Median line: fictitious line which follows and joins the* upper part of the character bodies in a text line.
- *Upper line: fictitious line which joins the top of* ascenders.
- *Lower line: fictitious line which joins the bottom of* descenders.
- *Overlapping components: overlapping components are descenders and ascenders located in the region of an ad*

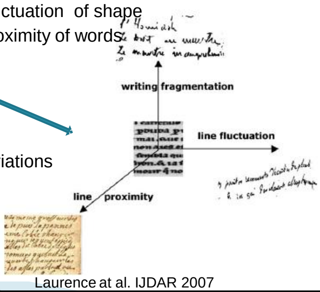


Problems in Document Sementation for Text Recognition

Line level:
- Fragmentation
- Fluctuation
- Proximity

Word level:
- Fragmentation of letters and words
- Fluctuation of shape
- Proximity of words

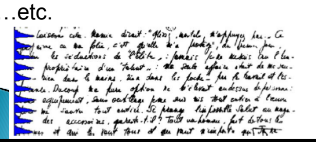
Sources of noise:
- Blotches
- Background intensity variations
- Transparency of paper
- Tears
- Scanning problems



Laurence at al. IJDAR 2007

Document Sementation for Text Recognition

- Projection-based methods
- Grouping methods: aggregating units in a bottom up strategy
- Smearing methods (horizontal smearing then bounding box decetion)
- Hough transform based methods
- …etc.



Super Resolution (SR) Based Number Plate (NP) Recognition

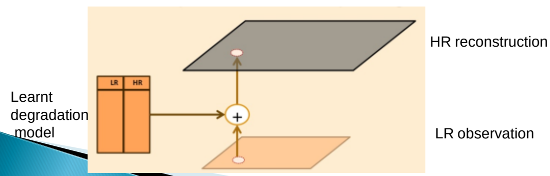
Problem: low resolution number plates in security videos

Solution: apply statistical image processing with the knowledge of what we expect to see (“***example based” super resolution***, ***image halucination***).

1. Learn low resultion – high resolution patch pairs by image examples
2. Retrieve high resolution patches from low resolution observation applying local constraints
3. Recognition: use reconstruction code statistics

Example-based SR

- Learn LR-HR image patch pairs by example images
- Build up a database from LR-HR pairs
- Replace LR patches with corresponding HR patterns also considering neighborhood fitting



Signature-based biometrics

user identification by password

Scanning → Analysis → Matching ⇄ Store

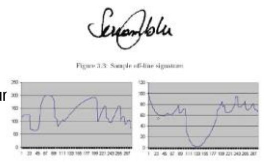
extracted features · outlier rejection · signature template

Match ← Decision → No match

two-class pattern recognition problem

Signature recognition

- Alignement
- Feature extraction
  - Baseline Slant Angle
  - Aspect Ratio
  - Normalized area of the signatur
  - Center of Gravity
  - Slope
  - Upper profile/lower profile
  - Etc.

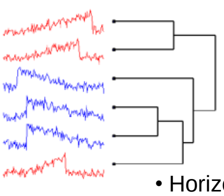


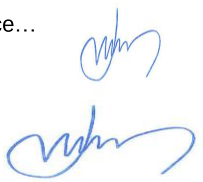
Upper profile/lower profile

- Comparison
  - Several types of metrics… (do not work alone) but
  - ***Dynamic Time Warping***
  - ***Hidden Markov Models*** can help…

Dynamic Time Warping…

- To find local correspondence…





• Horizontal non-linear stretching of objects to find the best matching
• Local gradient algorithms work well

The amount of information in archive documents…

Consumed by an average person on an average day

- corresponds to 100,500 words
- and 34 gigabytes
- newspapers, books, portable computer games, satellite radio, and Internet video,
- (information at work is not included!)

`How Much Information? 2009 Report on American Consumers, University of California, San Diego)`

Estimated number of books:

- 129,864,880.
- „at least until Sunday"

`(Google Books research, 2010)`

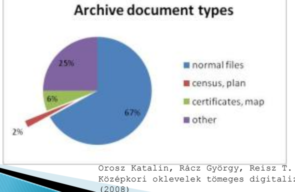
- What about old documents…?

What about achive paper documents?

- The number of archive pages (only in Hungary):

3 500 000 000 – over 3 billion!

- The number of archive pages recommended for digitization: 200 000 000 (5.7%)



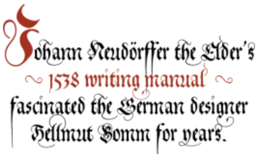
`Orosz Katalin, Rácz György, Reisz T. Csaba, Vajk Ádám, Véber János, Középkori oklevelek tömeges digitalizálása, Magyar Országos Levéltár, (2008)`

Aims of Digitization

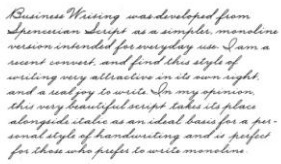
- To preserve information for future generations
- To make them analyzable for researchers
- To make them searchable for the public

- Central European Virtual Archives Network of Medieval Charters Project: … Digitization of medieval charters within the stocks of the participating archives…

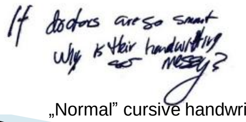
Handwriting styles



Fraktur handwriting



Cursive handwriting



„Normal" cursive handwriting

A page of a book of census of a Central-European city from 1771 (Veszprém County Archives)

Traditional OCR software products?

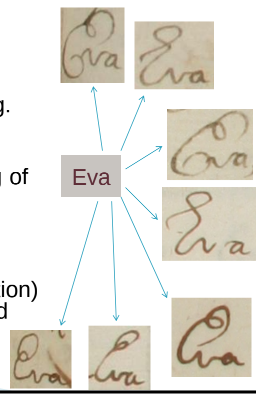
- FreeOCR,
- TOCR viewer,
- SimpleOCR,
- Abby FineReader,
- TOPOCR,
- …

simply do not work…archivists process information manually…

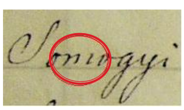
Typical Problems of Archive Cursive Handwriting

- The same letters have different appearances (e.g. „E" in Eva)
- The beginning and ending of letters can not be easily recognized

separation (segmentation) of letters is a (too) hard problem

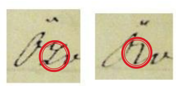


Typical Written Problems



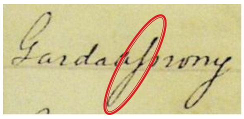
Broken line transforms "m" into "n" and "r"

Typical Written Problems



Different appearances of the same letter "z" in the same hand-writing (beginning of "Özvegy")

Typical Written Problems



Misspelling of the 7th letter which should look like the 8th letter.

Typical Written Problems

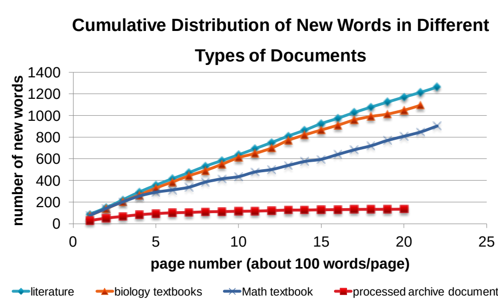
Similarities of different letters in the same hand-writing (beginning of "István", "János", "Sámuel")

Typical Written Problems

Word overlapping.

Consequences

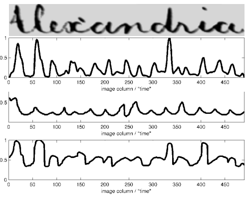
- Character-based recognition in several cases does not work.
- Is it worth trying word-based recognition – word spotting?
- What is the amount of word classes?



A continuously „learning system" seems to be reasonable, the amount of necessary annotation decreases exponentially from page to page in the archive document to be processed.

Global word shape based classification

- Tested descriptors of length 329
  - horizontal and vertical size and their ratio;
  - minimum, maximum, and average intensity;
  - average intensity derivatives;
  - upper profile; lower profile;
  - right profile; left profile;
  - center of gravity;
  - black-white transitions; black-white ratio;
  - black count;
  - black density;
  - image moments



- Tested classifiers: k-NN, Random Tree, Random Forest, Naive Bayes ect.
- Average performance is around (only) 50% recognition rate

Global word shape based classification

- **Global** word feature descriptors are
  - Sensitive to the individual (inter class) variations of word shape
  - Sensitive to extreme decorations
  - Sensitive to dirt and noise
  - are „ad-hoc"
- What about **local** feature descriptors in word spotting?
  - SIFT, SURF, FAST, … successfully applied to complex images
  - Invariant to transformations (rotation, scaling)

Local features for word spotting

- *Has it been already applied?*
- *Is scale invariance of descriptors important to be considered?*
- *Is rotation invariance of descriptors important to be considered?*
- *Is word structure (f.e. skeleton) itself proper to extract local features?*

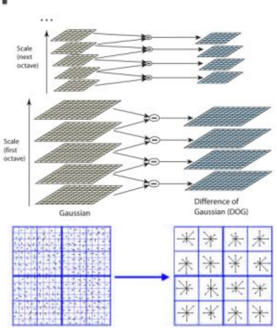
Existing solutions

- Lawrence Spitz: Using Character Shape Code for Word Spotting in Document Images (1995)
  - „SIFT-like" descriptor
  - Applied to Chinese symbols
  - *Not scale and rotation invariant*
- J. A. Rodríguez, F. Perronnin: Local Gradient Histogram Features for Word Spotting in Unconstrained Handwritten Documents. *Frontiers in Handwriting Recognition* (2008)
  - Gradient histogram descriptor in a moving window
  - DTW or HMM for classification
  - 80% hit rate for a low number of classes
  - *No information selection*
- Uchida, S.; Liwicki, M., Part-Based Recognition of Handwritten Characters, Frontiers in Handwriting Recognition (ICFHR), 2010 International Conference on, 545–550 (2010)
  - Tested and applied only for the 10 digits
  - SURF points without positions (*not real localization*)
  - Feature point votes for character class

More comprehensive overview is available in Czúni et al., CBMI2013

SIFT local descriptor

- Scale Invariant Feature Transform
  - Difference of Gaussian pyramid
  - Finding local extreme points (position, scale)
  - Leaving out low contrast and edge points
  - Finding the maximal gradient (for orientation invariance)
  - Setting the local coordinate system
  - Generating the descriptor vector
- Properties
  - Invariant to affine transformations (scaling, rotation, etc.)
  - Computationally expensive



[3] Lowe, D. G. Object Recognition from Local Scale-invariant Features *In Proceedings of the International Conference on Computer Vision 2* (1999)

1. Localize SIFT points and generate SIFT descriptors both in the query (q) and in the candidate words (c).
2. Normalize SIFT point positions by the physical size of the words.
3. Define a disk shape area around each feature point of the query (q): only candidate points (c) within this area will be compared.
4. Find the best two matching points

$$D(q_i,c_j)=\sqrt{\sum_{k=1}^{128}(q_i(k)-c_j(k))^2}$$

$$c_{i,min}=\min_{c_j} D(q_i,c_j)$$

$$c_{i,min2}=\min_{c_j} D(q_i,c_j) \text{ s.t. } c_j\neq c_{i,min}$$

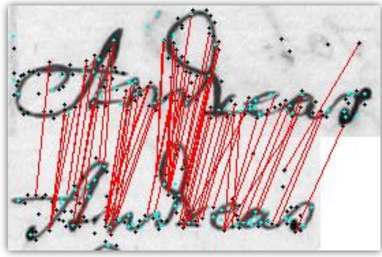
5. Apply a threshold to orientation difference
6. Constrain the uniqueness of the best matching point $\frac{D(q_i,c_{i,min})}{D(q_i,c_{i,min2})}<T_D$
7. Calculate the similarity value for the query and candidate words with the use of the matching points, rank candidates according to this similarity value:

$$S(Q,C)=\sum_{i=1}^{N}\left(\sqrt{255^2\cdot 128}-D(q_j,c_j)_{(q_j,c_j)\in M_{Q,C}}\right)$$

Advantages

- Scale and rotation invariance (in some degree)
- No need for preprocessing (e.g. binarization, slant correction, noise removal, morphology, etc.)
- No need for precise segmentation of words.
- The searching area is symmetrical around query points, contrary to most methods using DTW, where matching cannot go backwards.
- Stable in noisy environments: the algorithm can neglect most noisy points.
- Only extrema points in scale–space are considered: there is no need to correlate points with small information content.

Example for matching points

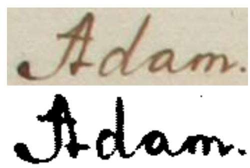


Experimental setup

- 22 manually annotated pages of the 177 with 1638 word images.
- 103 random query image compared to the remaining 1637 images
- 111 word classes
- most frequent word: 116 occurrence
- 68 words with only 1 occurrence
- SIFT (OpenSIFT, Lowe), SURF

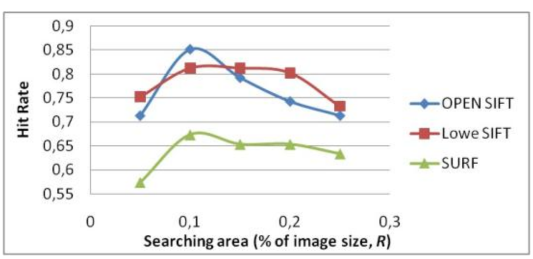
Preprocessing

- Segmentation - manually
- Noise-filtering
- Slant correction
- Word image resizing
- Binarization
- Skeletonization

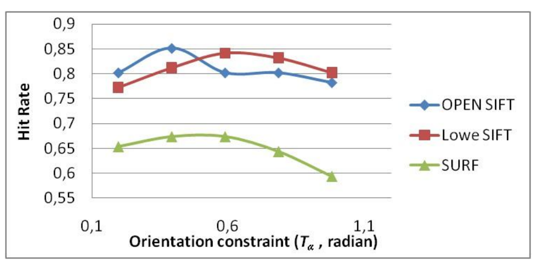


methods caused new problems… gave no real improvement

Effect of searching distance

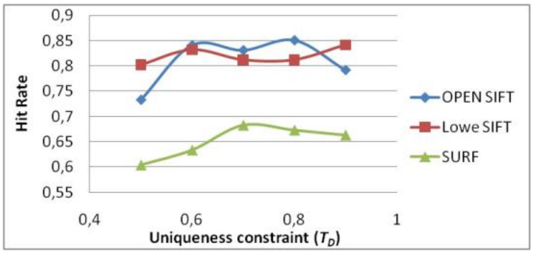


Effect of orientation constraint



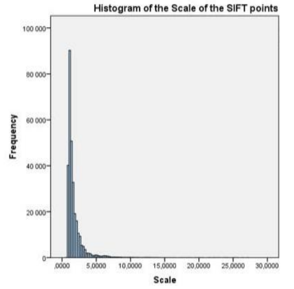
- Relativel stable performance of SIFT
- SURF is much behind

Effect of uniqueness constraint

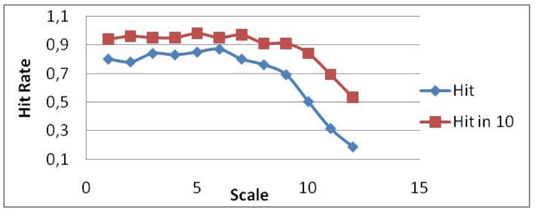


- Relatively stable performance of SIFT
- SURF is much behind

Distribution of Scale of over 300000 SIFT points from archive text

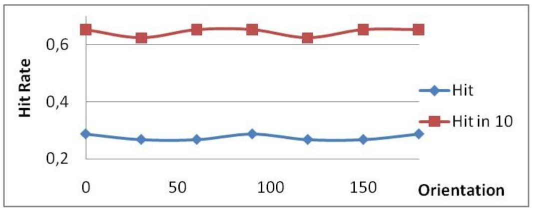


Effect of fixing the Scale



Scale information has relatively low importance in recognition

Effect of fixing the orientation



Constantly low performance tells us that, while written text is basically horizontal, but rotation invariance can't be neglected!

Analysis of results

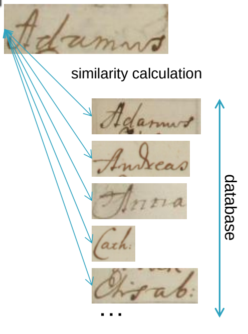
The list and images of mistaken recognitions from 101 random queries. Yellow words indicate classes of almost the same names.

Ground truth	Wrong recognition
Andreas	Ivannes
Anto	Anna
Catha	Cath
Cathar	Catharina
Eva	Anna
Filius	Viduus
Jannes	Ivannes
Joseph	Josephus
Julia	Julian
Math	Mich
Paulus	Paul
Sebastianus	Szephanus
Susana	Susanna
Vidua	Adam
Viduus	Filius

recognition error (in the test database) could be halved by grapheme processing

Sequential search

1. local feature extraction (SIFT)
2. calculating similarity value with the images of the database
3. searching the word with maximal similarity value

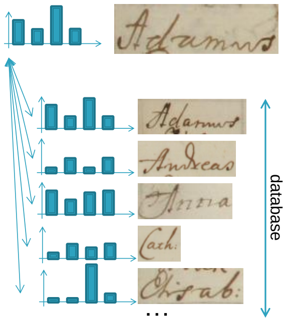


- the similarity calculation is slow
- long running time

Bag of Words (typical for SIFT)

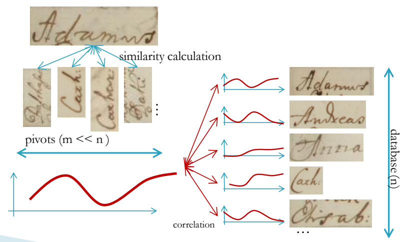
1. local feature extraction (SIFT)
2. create feature cluster histograms
3. calculating similarity values between histograms (eg. correlation)

– features are too sparse/similar
– poor recognition rate



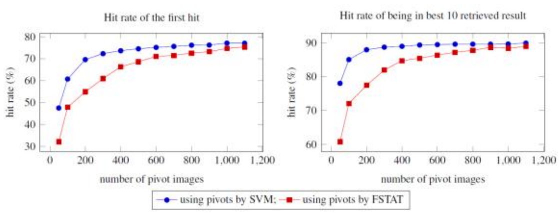
Pivot based searching

1. local feature extraction (SIFT)
2. create similarity values with the ***pivot images***
3. correlation calculation between the pivot similarity values („function") and the images of the database



Pivot based searching results

- hit rate is about 70-75 %
- 2-3 times faster searching depending on the size of the database
- pivot selection problem (SVM, FSTAT)



Conclusion

- **Not localized feature** descriptors are not proper for noisy archive handwriting recognition
- **SIFT based retrieval** can reach around **85% hit-rate** in case of cursive handwritten text with limited vocabulary
- Around 100% in the first 10 of the **result list** (manual correction is possible)
- **No need** for:
  - Preprocessing (e.g. binarization, slant correction, morphology)
  - Noise filtering
  - Precise segmentation
- **Rotation invariance** is more important than scale invariance
- Pivot based search can increas speed 2-3 times with small loss in retrieval rate
- Not all popular descriptors are adequate (SURF is faster but has significantly lower performance)

Thank you for your attention!

ACKNOWLEDGEMENTS : This research was supported by the Hungarian Government and the European Union and co-financed by the European Social Fund under project TÁMOP-4.2.2.C-11/1/KONV-2012-0004. László Czúni was supported by the Bolyai scholarship of the Hungarian Academy of Sciences.