Improving isolated and in-context classification of handwritten characters

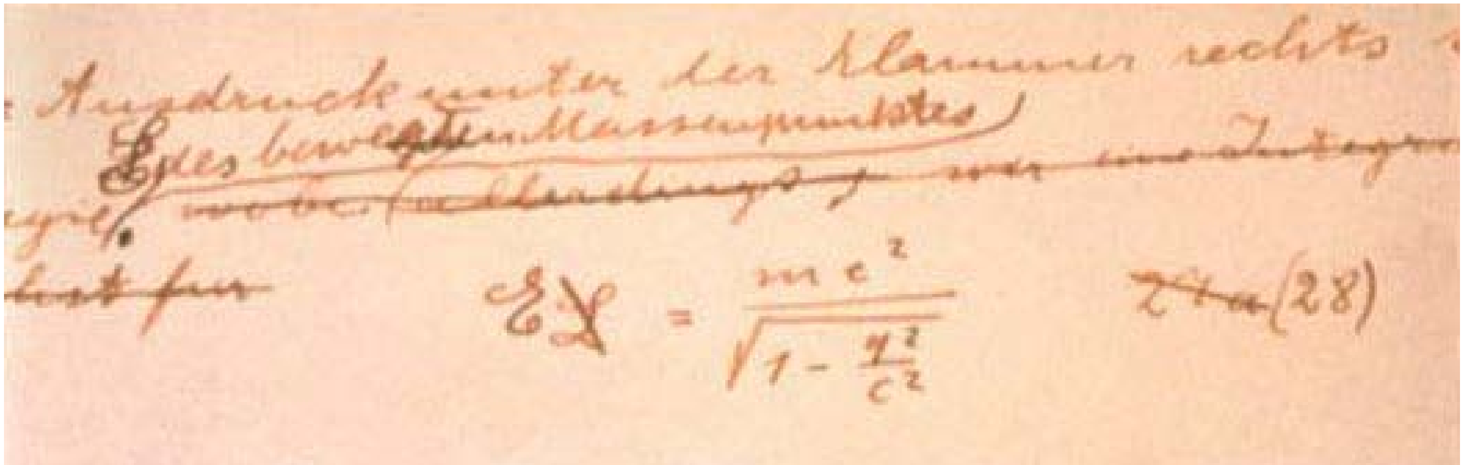
Vadim Mazalov and Stephen M. Watt

Joint Lab Meeting of ORCCA
Department of Computer Science
The University of Western Ontario

November 4, 2011

Western ORCCA

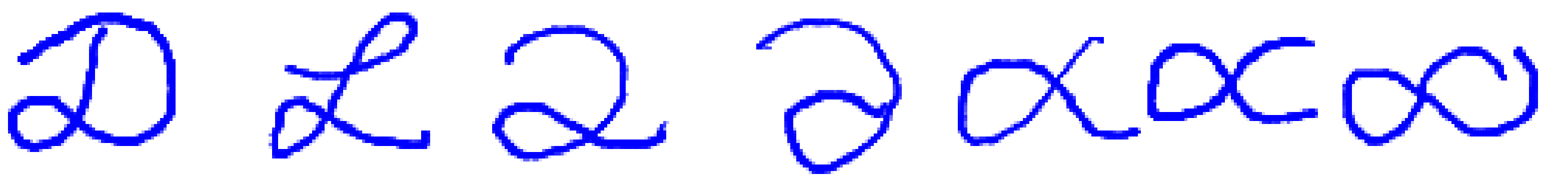
Pen-Based Math



- Input for CAS and document processing.
- 2D editing.
- Computer-mediated collaboration.

Pen-Based Math

- Different than natural language recognition:
  - 2-D layout is a combination of writing and drawing.
  - Many similar few-stroke characters.
  - Many alphabets, used idiosyncratically.
  - Many symbols, each person uses a subset.
  - No fixed dictionary for disambiguation.



Orthogonal Series Representation

- **Main idea**:
  Represent *c*oordinate curves as truncated orthogonal series.

- **Advantages**:
  - *Compact* – few coefficients needed
  - *Geometric*
    – the truncation order is a property of the character set
    – gives a natural metric on the space of characters
  - *Algebraic*
    – properties of curves can be computed algebraically
    (instead of numerically using heuristic parameters)
  - *Device independent*
    – resolution of the device is not important

Inner Product and Basis Functions

- Choose a functional inner product, e.g.

$$\langle f, g \rangle = \int_a^b f(t)g(t)w(t)dt$$

- This determines an orthonormal basis in the subspace of polynomials of degree *d.*

- Determine ϕ_i using GS on $\{1, t, t^2, t^3, \dots\}$.

- Can then approximate functions in subspaces

$$A(t) \approx \sum_{i=0}^{d} \alpha_i \phi_i(t) \qquad \alpha_i = \langle A(t), \phi_i(t) \rangle$$

First Look: Chebyshev Series

- Initially used Chebyshev series
  [Char+SMW ICDAR 2007].

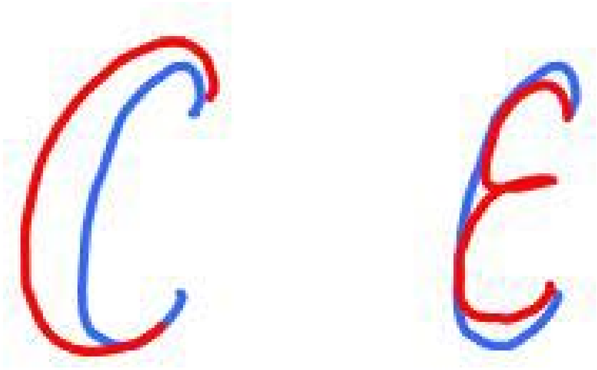
- Found could approximate closely
  (small RMS error) with series of order 10.

- Like symbols tended to form clusters.

Problems

- Want fast response –
  how to work while trace is being captured.

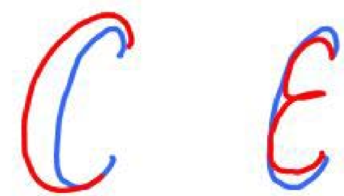
- Low RMS does not mean similar shape.



On-Line Series Coefficients

- Use Legendre polynomials P_i as basis on the interval $[-1,1]$, with weight function 1.
- Collect numerical values for $f(\lambda)$ on $[0, L]$.
  λ = arc length.
  L is not known until the pen is lifted.
- *As the sample points are collected*, numerically integrate the moments $\int \lambda^i f(\lambda) d\lambda$.
- After last point, compute series coefficients for f with domain and range scaled to $[-1,1]$.
  This uses a simple linear transformation of the moments.

Shape vs Variation



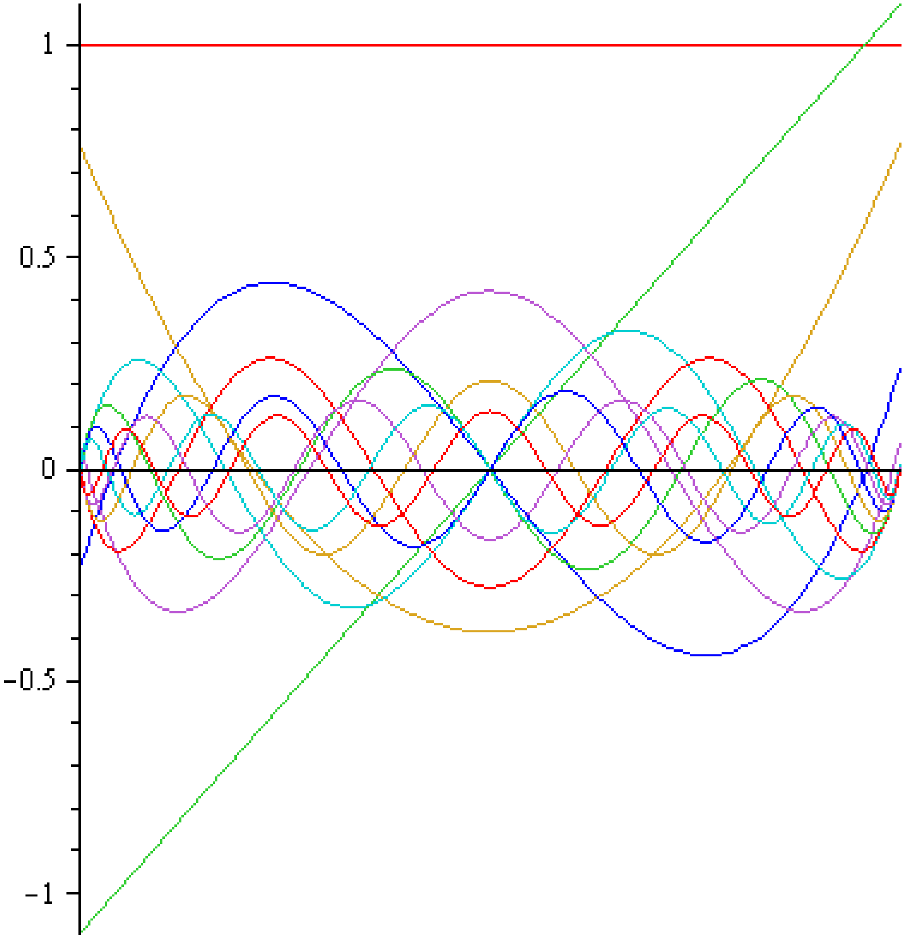
- The corners are not in the right places.
- Work in a jet space to force coords & derivatives close.
- Use a Legendre-Sobolev inner product

$$\langle f, g \rangle = \int_a^b f(t)g(t)dt + \mu_1 \int_a^b f'(t)g'(t)dt + \mu_2 \int_a^b f''(t)g''(t)dt + \cdots$$

- 1[st] jet space ⇒ set $\mu_i = 0$ for $i > 1$.
  - Choose μ_1 experimentally to maximize reco rate.
  - Can be also done on-line.

    [Golubitsky + SMW 2008, 2009]

Legendre-Sobolev Basis



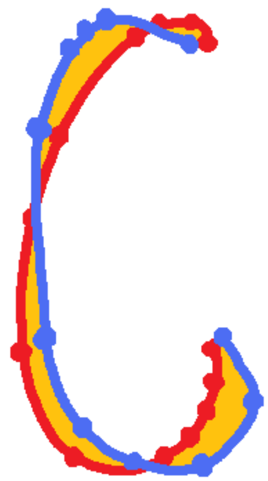
$$a = 0, b = 1, \mu = .125$$

Life in an Inner Product Space

- With the Legendre-Sobolev inner product we have
  - Low dimensional rep for curves (10 + 10 + 1)
  - Compact rep of samples ~ 160 bits [G+W 2009]
  - **>99% linear separability => convexity of classes**
  - A useful notion of distance between curves
    that is very fast to compute

Distance Between Curves

- **Elastic matching:**
- Approximate the variation between curves by some fn of distances between sample points.
- May be coordinate curves or curves in a jet space.



- Sequence alignment
- Interpolation ("resampling")

- **Why not just calculate the area?**
- This is very fast in ortho. series representation.

Distance Between Curves

$$\bar{x}(t) = x(t) + \xi(t) \qquad \xi(t) = \sum_{i=0}^{\infty} \xi_i \phi_i(t), \qquad \phi_i \text{ ortho on } [a,b] \text{ with } w(t) = 1.$$

$$\bar{y}(t) = y(t) + \eta(t) \qquad \eta(t) = \sum_{i=0}^{\infty} \eta_i \phi_i(t)$$

$$\rho^2(C, \bar{C}) = \int_a^b \left[\left(x(t) - \bar{x}(t) \right)^2 + \left(y(t) - \bar{y}(t) \right)^2 \right] dt$$

$$= \int_a^b [\xi(t)^2 + \eta(t)^2] dt$$

$$\approx \int_a^b \left[\sum_{i=0}^{d} \xi_i^2 \phi_i^2(t) + \text{cross terms} + \sum_{i=0}^{d} \eta_i^2 \phi_i^2(t) + \text{cross terms} \right] dt$$

$$= \sum_{i=0}^{d} \xi_i^2 + \sum_{i=0}^{d} \eta_i^2$$

Comparison of Candidate to Models

- Use Euclidean distance in the coefficient space.

- *Just as accurate* as elastic matching.
- *Much less expensive.*
- Linear in d, the degree of the approximation.
  < 3 d machine instructions (30ns) *vs* several thousand!
- Can trace through SVM-induced cells incrementally.

- Normed space for characters gives other advantages.

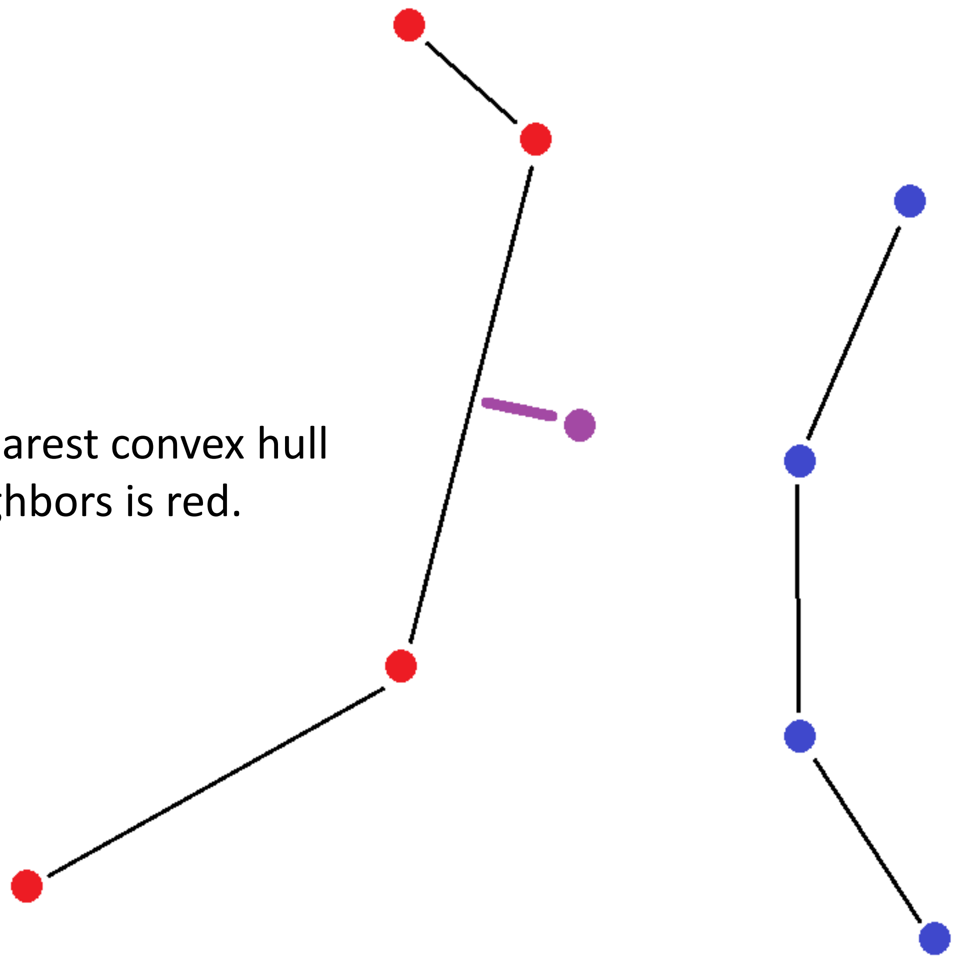
The Joy of Convex

- Linear separability ⇒ Convexity
  ⇒ Linear homotopies stay within a class

$$C = (1- t)\,A + t\,B$$

- Can compute distance of a sample to this line
- Distance to convex hull of nearest neighbours in class gives best recognition [Golubitsky+SMW 2009,2010]

Choosing between Alternatives



The nearest convex hull of neighbors is red.

Transformation-independent recognition

- Rotation- and shear-independent recognition can be achieved by computing invariant functions from the coordinates of a character.

- Then, similar to the coordinate functions, the invariants can be approximated with orthogonal polynomials and their coefficients used in recognition.

- We have shown in the past, that integral invariants can be a suitable descriptor of online handwritten characters.

The main contributions of this work

- Optimization of isolated character recognition by adjusting the
  - "jet scale" in the LS inner product for coordinate functions and integral invariants
  - degree of approximation of coordinate functions and integral invariants.
- An in-context rotation-invariant algorithm that yields substantially better results than isolated recognition and can be extended to other transformations.

Coordinate Functions

- The optimal μ for approximation of coordinate functions was found to minimize classification error.
- The original characters in the dataset were considered without any distortions.
- The characters were approximated with corresponding values of μ in the range from 0 to 0.2.

Complexity of Handwritten Characters

- We considered the possibility that the optimal value of μ might depend on the nature of the characters to be recognized.
- We took the notion of a sample's complexity as

$$\eta = \sum_{i=1}^{d} (X_i^{1/i} + Y_i^{1/i}), |X_i| \leq 1 \text{ and } |Y_i| \leq 1$$

  where X_i and Y_i are normalized coefficients of approximation of the sample with orthogonal polynomials
- The sample complexity function is derived from the fact that coefficients of higher degree are typically greater for "complex" characters – characters that contain large number of loops and/or amount of curvature.

In-context transformation-invariant recognition

- Recognition of distorted math symbols without context is sometimes impossible

$$\text{“}<\text{” vs } \angle, \text{ “}|\text{” vs “}/\text{”, “}\cup\text{” vs “}\subset$$

- It has been shown that n-grams provide useful semantic information in a mathematical setting.
- We use context to improve distortion-independent classification, taking advantage of the fact that samples written by a person typically exhibit similar degree of transformation.
- We consider the case of rotation (shear and other transformations may be handled similarly).

Experimental setting

- The testing set was represented in InkML format.
- It contains 50,703 characters in 242 classes.
- Class labels incorporated the number of strokes (therefore, single-stroke and double-stroke “7” were considered as different classes).

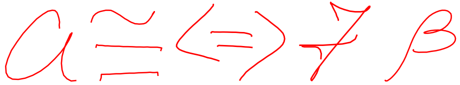


Figure: Characters from the training dataset

- The model was trained with non-transformed samples.
- For the recognition phase, sequences of n characters were taken from the dataset and each sequence was rotated by a random angle $\gamma \in [-\beta, \beta]$.

Coordinate functions

- The figure shows the error of recognition using coefficients of X and Y.

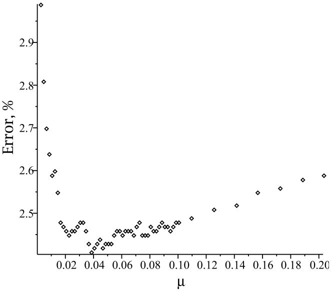
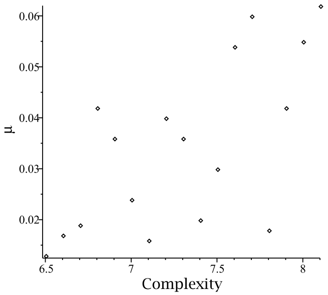


Figure: Recognition error of non-transformed characters for different values of μ

- $\approx 2.4\%$ error is reached for $\mu = 0.04$.

Optimal μ for characters with different complexities

- We found that the optimal μ is not strongly correlated with the complexity of characters.



- Results of Spearman and Kendall tau-a correlation tests between complexity and μ are respectively: $\rho_{\mu,\eta}(13) = 0.52, p = 0.047$ and $\tau_{\mu,\eta}(13) = 0.38, p = 0.053$.

I_0

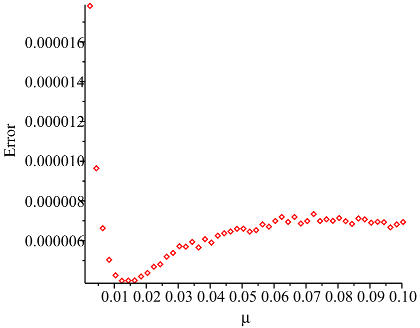
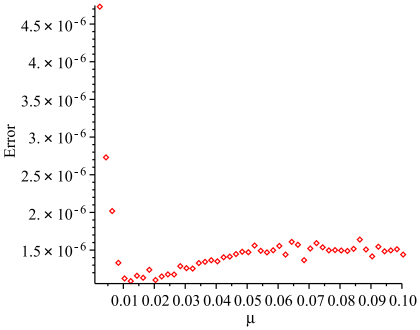


Figure: Average maximum error in coefficients of I_0 depending on μ

I_1



(a)

Figure: Average maximum error in coefficients of I_1 depending on μ

Degree of Approximation

Table: Recognition error (Rec.Err.), maximum approximation error (Max.Err.) and average relative approximation error of coordinates (Avg.Err.) for different degrees of approximation d, $\mu = 0.04$

d	9	10	11	12	13	14	15
Rec.Err. %	2.57	2.49	2.46	2.43	2.44	2.45	2.46
Max.Err.	707	539	539	484	475	494	500
Avg.Err.($\times 10^{-3}$)	1.9	1.6	1.4	1.2	1.1	1.0	1.2

We find degree 12 to be the optimum for recognition of symbols in our collection.

In-Context Classification: Parameters

There are 3 parameters that in-context recognition rate can depend on:

- Number p of closest classes in computation of error likelihood.
- Rotation angle.
- Size n of the set of characters.

Evaluatoin of p

- We fixed the parameter $n = 3$.
- Performed classification for values p of 2, 3 and 4.
- We found that p has almost no effect on recognition error.
- We took $p = 3$ and continued the experiments.

Evaluation of n and the rotation angle

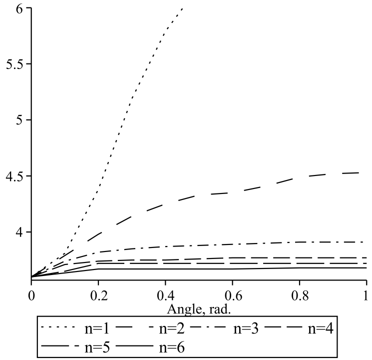


Figure: Recognition error (%) for different size of context n and different angles of rotation (in radians)

A significant reduction in error is achieved between $n = 1$ and $n = 3$.

Contributions of the presented work

- The optimal range of values for the jet scale for coordinate basis functions.
- A study that demonstrates that this optimal value of μ, to a first approximation, does not depend on the complexity of the characters tested.
- The optimal values for the jet scale for the integral invariants I_0 and I_1, used in rotation-independent recognition.
- The optimal degree of the approximating series.
- An algorithm for orientation-invariant in-context classification by considering characters in groups.