A HANDWRITTEN DIGIT RECOGNITION ALGORITHM USING TWO-DIMENSIONAL HIDDEN MARKOV MODELS FOR FEATURE EXTRACTION

Jay Wierer and Nigel Boston

University of Wisconsin - Madison
Department of Electrical and Computer Engineering

ABSTRACT

We propose a handwritten digit recognition algorithm that uses 4x4 2D hidden Markov models to extract basic features from an unclassified image. The novel idea given here is that we use powerful techniques from the emerging mathematical fields of tropical geometry and algebraic statistics to determine parameters for the model. The distance between the unclassified images and prototypes is calculated in stages, where estimates of the distance become finer as obviously distant prototypes are discarded from the pool of possible K-nearest neighbors. Our algorithm achieves a 95.51 percent recognition rate with zero rejection on the MNIST database of handwritten digits.

Index Terms— Handwriting recognition, character recognition, Hidden Markov models, feature extraction

1. INTRODUCTION

Handwritten digit recognition is an area of pattern recognition that has seen much active research in the past decade. Various approaches have been taken to push recognition rates to those near human performance. Most approaches involve statistically based methods, due to their relative ease of implementation as compared to semantic methods. Some such methods include K-nearest-neighbor classification, convolutional neural nets [1], shape matching [2], support vector machines [3], tree classifiers [4], tangent distance [5], and hybrid methods based on outputs of multiple classifiers [6].

Graphical models such as the hidden Markov model (HMM) have also been used in handwritten character recognition. Agazzi et al [7] employ a quasi-two-dimensional method called the planar hidden Markov model (PHMM) for recognizing severely degraded text. Levin and Pieraccini [8] further the idea of the PHMM by adding a dynamic planar warping (DPW) algorithm to align a two-dimensional reference image with an elastically distorted test image. Merialdo et al [9] propose a two-dimensional Viterbi algorithm for the two-dimensional hidden Markov model to be used for character segmentation and recognition.

We propose to use a genuinely two-dimensional hidden Markov model (2D HMM) [10] to extract features from unclassified digit images, which are drawn from the MNIST database (available at http://yann.lecun.com/exdb/mnist/). Distances between unclassified digit images and prototypes are computed hierarchically, that is, distance computations become finer as the algorithm enters later stages and as prototypes with large distances from the unclassified image are removed from consideration.

This paper is the first we know to use powerful techniques from the emerging mathematical fields of tropical geometry and algebraic statistics [11, 12] to perform character recognition. Other fields, such as computational biology, biostatistics, and genomics, have incorporated these algebraic tools to tackle problems in their respective areas. We hope that this paper will inspire others to experiment with these novel techniques for statistical analysis.

2. TWO-DIMENSIONAL HIDDEN MARKOV MODELS (2D HMMS) AND NEWTON POLYTOPES

2.1. Two-dimensional HMM

The two-dimensional hidden Markov model has been largely avoided in the literature because of its computational complexity in training its parameters (exponential in n^2). Merialdo et al [9] provide a 2D Viterbi algorithm which claims to achieve subexponential computational complexity. Li et al [13] propose a 2D HMM image classification algorithm using both the 2D Viterbi and EM algorithms. We propose here an algorithm for 2D HMM feature extraction, based on the Newton polytope, which we describe later in this section.

The two-dimensional hidden Markov model that we employ [10] has the following properties:
1. Each hidden state $X_{i,j}$ depends on its "past" only through its immediate left and immediate top neighboring states $X_{(i-1)_n,j}$ and $X_{i,(j-1)_n}$, where $(x)_n = x \pmod n$.
2. Each observation bit $Y_{i,j}$ depends only on its corresponding hidden state $X_{i,j}$.
3. All random variables are binary variables (having values 0 or 1).
4. We denote $P(X_{i,j} = m|X_{(i-1)_n,j} = k, X_{i,(j-1)_n} = l) = a_{k,m}a_{l,m}$ and $P(Y_{i,j} = l|X_{i,j} = k) = b_{k,l}$.

For a fixed observation $Y = [y_{i,j}]$,

$$\begin{aligned} f_Y &= P(Y) \\ &= \sum_{X \in X_n} \prod_{(i,j)} a_{x_{(i-1)_n,j},x_{i,j}} a_{x_{i,(j-1)_n},x_{i,j}} b_{x_{i,j},y_{i,j}}, \end{aligned} \tag{1}$$

where $X_n = \{0,1\}^{n \times n}$. Here we assume that $a_{00} = a_{10}$, $a_{01} = a_{11} = \sqrt{1 - a_{00}^2}, b_{00} = 1 - b_{01}, b_{10} = 1 - b_{11}$. This unfortunately limits the domain of allowable parameters $\lambda = [a_{00}, a_{01}, a_{10}, a_{11}, b_{00}, b_{01}, b_{10}, b_{11}]$, however, this will save us computation in calculating the Newton polytope described below.

Figure 1 gives a graphical representation of the 2D HMM.

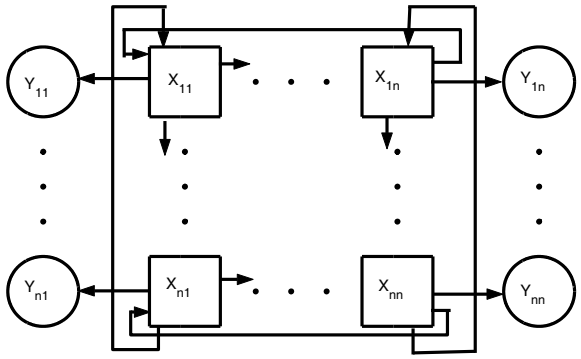


Fig. 1. Graphical representation of an n-by-n 2D HMM

2.2. Newton polytopes of 2D HMMs

In the budding mathematical fields of algebraic statistics and tropical geometry [11], the Newton polytope, which is given below, plays an important role in statistical inference.

Given a polynomial map $f : \mathbf{R}^m \to \mathbf{R}$ which can be expressed as

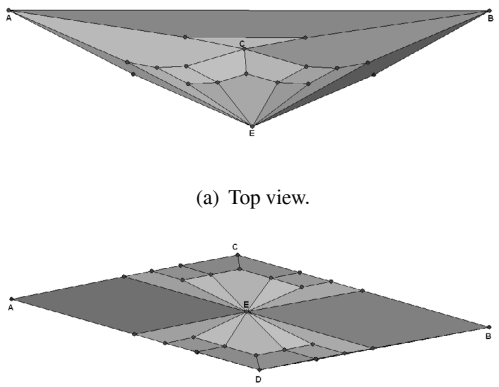
$$f = \sum_{i=1}^{q} c_i x_1^{\nu_{i,1}} x_2^{\nu_{i,2}} ... x_m^{\nu_{i,m}},$$

the **Newton polytope** Newt(f) of f is defined as the convex hull of the point set $N = \{(\nu_{i,1}, \nu_{i,2}, ..., \nu_{i,m}) | i = 1, ..., q\}$, i.e. Newt($f$) = conv($N$).

For a given observation $Y \in X_n$ with a fixed number k of ones, we find [10] that the Newton polytope is three-dimensional and conjecture that the polytope has $O(n^2)$ vertices.

Consider the 4x4 observation grid

$$Y = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$



(a) Top view.

(b) Front view.

Fig. 2. Newton polytope for a 4-by-4 observation with 31 vertices

Figure 2 shows the Newton polytope Newt(Y), which has the maximal number of vertices for all observations in X_4.

The usefulness of the Newton polytope has to do with its vertices, namely, the vertices of the Newton polytope of a polynomial map expressing the marginal probability (such as f_Y above) of an observation correspond to the hidden states that maximize the probability for a given set of parameters λ.

3. HANDWRITTEN DIGIT RECOGNITION ALGORITHM

3.1. Feature extraction using Newton polytopes

We compute features by employing the Newton polytopes arising from the 2D HMM. For each feature k we hope to extract in a 4x4 window – vertical line ($\updownarrow$), horizontal line ($\leftrightarrow$), diagonal lines ($\nearrow$ and $\searrow$), all zeros (**0**), all ones (**1**) – we train a 2D HMM for each representative feature grid $Y = Y^{(k)}$ and select the parameter vector λ which maximizes $P(Y|\lambda)$ by using the tropical approximation

$$g_Y = \min_{X \in X_n} \sum_{(i,j)} [s_{x_{i,j},x_{(i+1)_n,j}} + s_{x_{i,j},x_{i,(j+1)_n}} + t_{x_{i,j},y_{i,j}}], \tag{2}$$

where $s_{.,.} = -\log(a_{.,.})$ and $t_{.,.} = -\log(b_{.,.})$.

Let $\gamma = -\log(\lambda)$ and v_X be the vector of powers of λ in the monomial of f_Y corresponding to X. We can rewrite g_Y as the following:

$$g_Y = \min_{X \in X_n} \gamma \cdot v_X = \min_{v \in V(\text{Newt}(Y))} \gamma \cdot v, \tag{3}$$

where V(Newt(Y)) denotes the vertex set of Newt(Y) and the rightmost term is due to the fact that non-vertices $v_X \in$ Newt(Y) will not minimize the inner product.

Let A be the collection of features. After selecting a parameter vector $\gamma^{(k)} = -\log(\lambda^{(k)})$ for each feature k, we compute features k^* for a 4x4 windowed image I by the following:

$$k^* = \arg\min_{k \in A}\{\min_{v \in V} < v, \gamma^{(k)} >\}, \tag{4}$$

where V is the vertex set of Newt(I).

One note of caution: the 2D HMM is invariant to translations and rotations by multiples of 90 degrees, hence we need additional information to distinguish between horizontal ($\leftrightarrow$) and vertical ($\updownarrow$) lines and diagonals ($\nearrow$ and $\searrow$). However, a simple variance calculation can be set up to account for this situation. In the case where the variance calculation does not distinguish one from the other, we denote this the "unknown" feature (given by $\times$).

To calculate the distance between features $d_f(x, y)$, we use the following expression:

$$d_f(x,y) = \begin{cases} 0, & x = y \\ 1, & x = \mathbf{0}, y = \mathbf{1} \\ \frac{1}{4}, & x \in \{\leftrightarrow, \updownarrow, \nearrow, \searrow\}, y = \times \\ \frac{1}{2}, & \text{otherwise} \end{cases}$$

3.2. Hierarchical distance algorithm

Following Simard et al [5], at each stage k of the algorithm we keep a pool $\mathcal{P}_k$ of prototypes which potentially contain the K-nearest neighbors of the pattern P. For each prototype $P_i \in \mathcal{P}_k$, we compute the distance $D_i^{(k)}$ between P and P_i. We also compute the class label L_k and confidence score C_k for each stage. The algorithm proceeds to the next stage if C_k is below a certain threshold T_k, and we put the N_k best prototypes (according to $D_i^{(k)}$) in $\mathcal{P}_k$ into $\mathcal{P}_{k+1}$. If $C_k \geq T_k$, we stop the algorithm and declare L_k as the classification for P. The 28x28 images given in the MNIST database are centered in a 32x32 zero-padded matrix for easier subsampling. We use 8x8 subsampled Euclidean distance as $D^{(1)}$, 32x32 full Euclidean distance as $D^{(2)}$, 8x8 (each 4x4 window without overlap) feature distance as $D^{(3)}$, and 15x15 (4x4 windows overlapping by 2 pixels horizontally and vertically) feature distance as $D^{(4)}$.

4. RESULTS

We used the first 50000 images in the MNIST training set as prototypes, reserving the final 10000 images for validation. In addition, the MNIST database contains 10000 images for testing the algorithm. The validation set was used for determining the constants T_k, N_k that maximize the recognition rate.

With $N_1 = 50000, N_2 = 5000, N_3 = 20, N_4 = 15, T_1 = 10, T_2 = 3, T_3 = 3$, we achieve 95.51 percent recognition rate with zero rejection on the MNIST testing database.

5. CONCLUSION

The primary reason that the number of prototypes kept in the pool $\mathcal{P}_3$ is so much smaller than those kept in $\mathcal{P}_2$ is that, even though the Newton polytopes are conjectured to have $O(n^2)$ vertices, the process of determining those vertices takes much more computation. The current process of determining (approximate) Newton polytopes is as follows:
1. First calculate the "essential" vertices (see [10]).
2. Compute a second layer of vertices which generally appear for most observations Y.
3. Determine additional vertices by changing each vertex grid by one pixel (n^2 different grids for each vertex grid) and compute the convex hull of the resultant expanded vertex set.
4. Mark all "new" vertices, return to step 3, and iterate either until there are no new vertices or until a certain count (say, five) has been reached.

A subsequent experiment using 3x3 Newton polytopes for feature extraction is currently being conducted. We suspect that the process of computing Newton polytopes will be much quicker in this case, since we have pre-calculated the Newton polytopes for all 26 orbits in X_3 [10], in which case we may use a look-up table to determine the Newton polytopes for each prototype.

We would also like to eventually use 28x28 Newton polytopes on the entire image, which would require for us to be able to calculate the Newton polytope efficiently and expediently. Additionally, we would have to find a better approximation for the probability $P(Y)$ than that given by Equation 2.

In all, this novel method using a truly 2D hidden Markov model for feature extraction performs quite well on a standard handwritten digit database, and we can only imagine that our results will improve as we learn more about the powerful tools afforded us by tropical geometry and algebraic statistics.

6. REFERENCES

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, November 1998.

[2] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," in *IEEE Trans. Pattern Analysis and Machine Intelligence*, April 2002, vol. 24, pp. 509–522.

[3] D. DeCoste and B. Schölkopf, "Training invariant support vector machines," *Machine Learning*, vol. 46, pp. 161–190, 2002.

[4] Y. Amit, D. Geman, and K. Wilder, "Joint induction of shape features and tree classifiers," *IEEE Trans. on*

Pattern Analysis and Machine Intelligence, vol. 19, pp. 1300–1305, November 1997.

[5] P. Simard, Y. Le Cun, J. Denker, and B. Victorri, "Transformation invariance in pattern recognition - tangent distance and tangent propagation," *Lecture Notes in Computer Science*, vol. 1524, pp. 239–274, 1998.

[6] K. Chellapilla, M. Shilman, and P. Simard, "Combining multiple classifiers for faster optical character recognition," *International Workshop Document Analysis Systems*, vol. 3872, pp. 358–367, 2006.

[7] O. Agazzi, S-s. Kuo, E. Levin, and R. Pieraccini, "Connected and degraded text recognition using planar hidden Markov models," *Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE International Conference on*, vol. 5, pp. 113–116, 1993.

[8] E. Levin and R. Pieraccini, "Dynamic planar warping for optical character recognition," *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, vol. 3, pp. 149–152, 1992.

[9] B. Merialdo, S. Marchand-Maillet, and B. Huet, "Approximate Viterbi decoding for 2D-hidden Markov models," *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on*, vol. 6, pp. 2147–2150, 2000.

[10] J. Wierer and N. Boston, "Newton polytopes of two-dimensional hidden Markov models," *Experimental Mathematics*, 2007, accepted for publication.

[11] L. Pachter and B. Sturmfels, "Tropical geometry of statistical models," *Proceedings of the National Academy of Sciences*, vol. 101, pp. 16132–16137, 2004.

[12] L. Pachter and B. Sturmfels, Eds., *Algebraic Statistics for Computational Biology*, Cambridge University Press, 2005.

[13] J. Li, A. Najmi, and R. Gray, "Image classification by a two dimensional hidden Markov model," *IEEE Transactions on Signal Processing*, vol. 48, pp. 517–533, February 2000.