The Geometry of the Manifold of an Image Class and Its Application to Classification

Erik G. Miller

Artificial Intelligence Laboratory
Massachusetts Institue Of Technology
Cambridge, Massachusetts 02139

http://www.ai.mit.edu

The Problem: In [3], we address the classic problem of handwritten character recognition. That is, given some class-labeled training examples of images of handwritten characters, identify the class of each of a set of test examples to the greatest possible accuracy. For example, when given a particular image of the character "8", identify it as such. On isolated single character test images, success rates of over 99 percent have been achieved on standard data sets, so one may question the need to continue addressing this problem. However, there are several good reasons to continue to work on this problem.

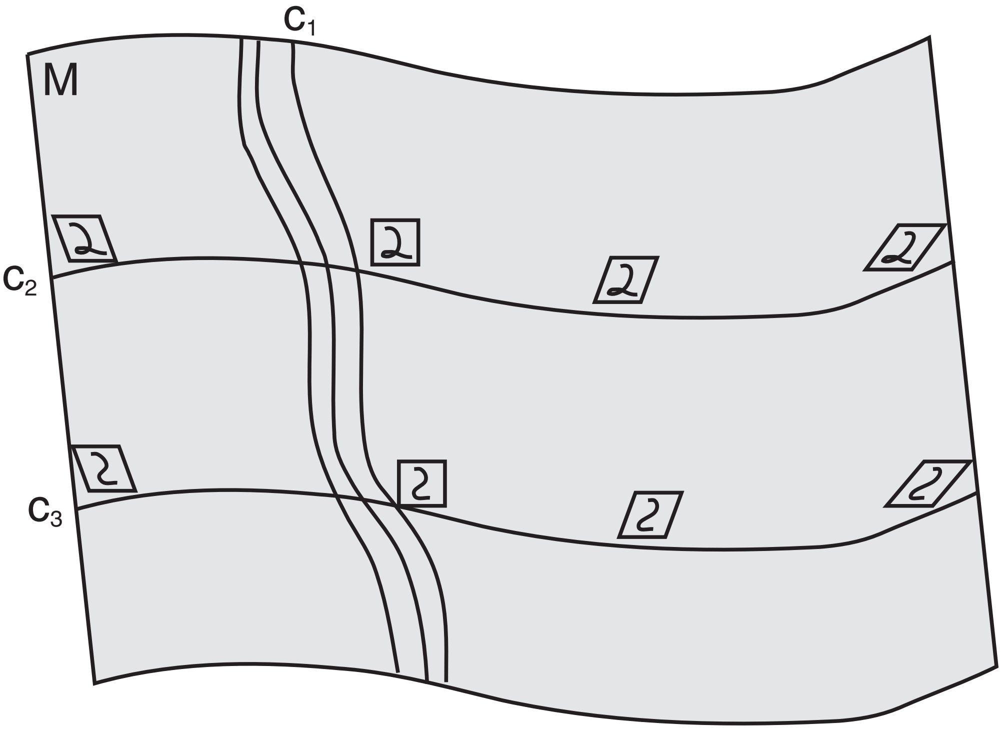

Figure 1: A diagram of the high-dimensional manifold of "2's", showing the foliation into style and transform. The curve C_1 and the curves approximately parallel to it represent the variation in style of the character. Each vertical curve represents a certain degree of transformation of the character. The curves labelled C_2 and C_3 show equivalence classes of characters with the same style, but at different transforms. The *canonical* element of each of these classes (the one closest to the empirical mean) is found at the intersection of their defining curve with the curve C_1. Roughly speaking, during testing, a new image is transported along the manifold until it reaches C_1, at which point it is compared with the canonical version of the style of "2" found at that point on the manifold. Hence, this procedure is essentially a form of non-linear projection.

Motivation: First, the most successful character recognizers have typically used a large training set, (6,000 examples per character), and furthermore have artificially augmented this training set by creating extra examples that are slight variations of the actual training data. In these cases, the training sets have grown in size to 60,000 examples per character. One goal is to greatly reduce the number of training examples without dramatically reducing the performance. If we are successful in this endeavor, then the application of these methods to other classification problems should be greatly facilitated, since not as much training data will be needed.

Second, many methods use significant domain specific knowledge, making it difficult to transfer these methods to other visual applications. While our method uses particular characteristics of the visual world (i.e. projective geometry), it uses no information specific to the classification of characters, and hence can be applied to other problems in visual classification.

Previous Work: The method of "tangent distance" [1] shares with our work the assumption that a class forms a manifold in image space. However, the ability of the algorithm to generalize depends upon the size of the neighborhood of a point in which the manifold is approximately flat. Since this neighborhood is small in the case of handwritten characters, a large number of training examples must be used to obtain good performance. [2] worked on separating "style" and "content" of characters in a factor model of characters. The chief difference from our work is that their model was bilinear, and ours is non-linear.

Approach: We model the variation in each class of character as depending on three factors, the "style" of the character, the "transform" of the character from an empirically defined canonical position, and independent pixel noise. The total probability of a character, given a particular model, is then given as the product of two probabilities, one that is independent of the transform component of the character, and one which is only a function of the transform.

In a non-linear projection step, a test character is transported along the equivalence class contours of the space (shown by the curves C_2 and C_3 in the figure). When the character is as close as possible to the canonical sub-manifold (represented by the curve C_1), a distance (or negative log likelihood) can be computed between the tranported character and the canonical sub-manifold.

The other component of the probability is arrived at by assigning a probability density to the affine transform used to do the non-linear projection in the previous step. For example, if a large rotation were used to make a "6" look like a "9", we would intuitively assign a low probability to such a transform, since we rarely need to rotate characters 180 degrees before they are recognized. Thus, for test character C, model M, and a transform $\mathcal{T}$ that transports the original C to the most canonical possible version of that style C_{canon}, we have:

$$P(C|M) = P(C_{canon}, \mathcal{T}|M) = P(C_{canon}|\mathcal{T}, M) * P(\mathcal{T}|M) \tag{1}$$

There are two benefits to performing this non-linear affine transport. First, it allows a denser modeling of the styles of a character with a small number of training examples. This is a basic characteristic of any factoring scheme. Second, by sharing the density used in the second factor above ($P(\mathcal{T})|M)$) between different character models, we hope to get a good density estimator for a character from just a few (or even just one!) examples.

Difficulty: One of the primary difficulties with this problem is that it is computationally intensive. Finding the optimal transform of a test character requires an on-line optimization, and this must be repeated for each model to be tested. Currently, this takes up to a minute per test character, even with only ten models. Clearly, a speed up in this performance would be desirable.

Impact: By sharing an understanding of transforms across models, we hope to significantly reduce the amount of training data required to develop models for new characters, and even other types of objects.

Future Work: Ultimately classifiers should do better by sharing additional features, other than just the densities on affine transforms. Searching for good sharable features will be a focus of future work.

Research Support: This work was supported by a fellowship from Microsoft Corporation.

References:

[1] P. Simard and Y. LeCun and J. Denker. Efficient Pattern Recognition Using a New Transformation Distance. *Neural Information Processing Systems 5*. Morgan Kaufmann, 1993.

[2] J. Tenenbaum and W. Freeman. Separating Style and Content. *Neural Information Processing Systems 9*. MIT Press, 1997.

[3] E. Miller and N. Matsakis and P. Viola. Learning from One Example Through Shared Densities on Transforms. *Computer Vision and Pattern Recognition, 2000*. IEEE, 2000.