Style-consistency in isogenous patterns

Prateek Sarkar
Xerox Palo Alto Research Center
Palo Alto, CA 94304, U.S.A
psarkar@parc.xerox.com

George Nagy
Rensselaer Polytechnic Institute
Troy NY 12180, U.S.A.
nagy@ecse.rpi.edu

Abstract

In many applications of pattern recognition, patterns appear in groups (fields) that have a common origin. For example, a printed word is a field of character patterns printed in the same font. A common origin induces consistency of style among features measured on patterns. In the presence of multiple styles, the features of co-occurring patterns are statistically dependent through the underlying style. Modeling such dependence among constituent patterns of a field increases classification accuracy. Effects of style consistency on the distributions of field-features (concatenation of pattern features) are modeled by hierarchical mixtures. Each field derives from a mixture of styles, while within a field a pattern derives from a class-style conditional mixture of Gaussians. An optimal (least error) style-conscious classifier processes entire fields of patterns rendered in a consistent but unknown style, based on the model. In a laboratory experiment style-conscious classification reduced errors on fields of printed digits by nearly 25% over singlet classifiers. Longer fields favor our classification method, because they furnish more information about the underlying style.

Keywords: Style, style consistency, style-conscious classification, isogenous patterns, field classification, mixture models

1 Introduction

Applications of pattern recognition often require the classification of groups of patterns having a common origin (*isogenous patterns*). For example, the character (bitmap) patterns in the image of a printed word are isogenous. They represent the same font, and are acquired by the same mechanism (print quality, degradation, scanner parameters). We use the term *field* to refer to such a group of isogenous patterns.

Patterns exhibit traits of their origin, thereby inducing *styles*. Thus there are different styles of printed character patterns (Figure 1), handwritten characters (owing to writers, and writing and digitizing devices), and speech patterns (speaker, recording environment, recording device). Though patterns may be produced in a multitude of styles, patterns in a field exhibit consistency of style because they are isogenous. Such consistency induces statistical dependence between feature-measurements on different patterns in a field. Modeling inter-pattern feature dependence can improve classification of patterns [Sar00].

We present a probabilistic model for style consistency in a field and demonstrate how the model can be used for improving classification accuracy. We explain the style-consistency model in Section 2. We discuss style conscious classification, and estimation of the model in Sections 3 and 4. In Section 5 we describe laboratory experiments on classification of printed numerals.

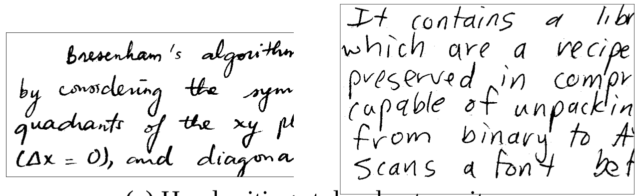
2 Probabilistic model of style consistency

Let $\mathbf{x}=(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_L)$ represent feature-measurements on L patterns in a field. The concatenation of pattern-feature vectors ($\boldsymbol{x}_l$) is called the *field-feature* vector ($\mathbf{x}$).

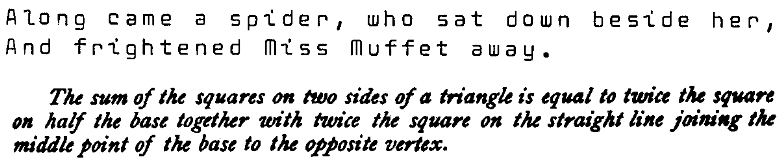
Let c_l represent the identity (*pattern-class*) of the l^{th} pattern of the field. The *field-class*, $\mathbf{c}=(c_1,\ldots,c_L)$, is the concatenation of pattern-classes. $p(\mathbf{x}|\mathbf{c})$ denotes the field-class conditional field-feature probability.

In the presence of multiple styles, style-consistency in a field induces statistical dependence among different patterns in a field. In *singlet* models such inter-pattern-feature dependence is not modeled, and the field-class conditional field-feature probability can be expressed as a product:

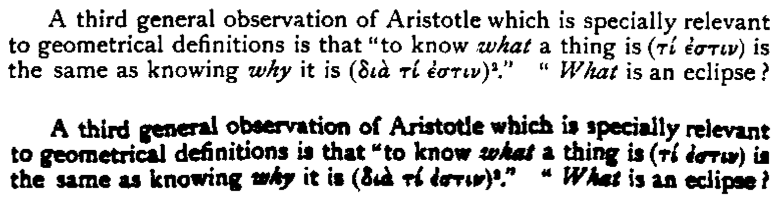
$$\begin{aligned} p(\mathbf{x}|\mathbf{c}) &= p(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_L|c_1,\ldots,c_L) \\ &= \prod_{l=1}^{L} p(\boldsymbol{x}_l|c_1,\ldots,c_L) \\ &= \prod_{l=1}^{L} p(\boldsymbol{x}_l|c_l) \end{aligned} \quad (1)$$



(a) Handwriting styles due to writers



(b) Printing styles (typeface, typesize)



(c) Styles due to printing, scanning mechanism

Figure 1. Fields of handwritten and printed characters in different styles

The final simplification in (1) is based on the assumption that the l^{th} pattern-feature, $\boldsymbol{x}_l$, depends on the class of the l^{th} pattern but is independent of all the other pattern-classes. This assumption is violated, for instance, by ligatures in handwriting and print (the shape of "i" is different in "six" and "fix"), and co-articulation in speech. In speech recognition, context trees model feature densities in the context of the field-class [Jel97]. We shall adopt this simplification, although it is not central to the discussion of style-consistency, and may be avoided by more complex style-consistency models.

In the presence of styles, the pattern-class conditional pattern-feature probability is a mixture density induced by different styles of patterns. For K styles, $1, \dots, K$

$$p(\boldsymbol{x}_l | c_l) = \sum_{k=1}^{K} \alpha_k p(\boldsymbol{x}_l | k, c_l)$$

where α_k is the probability of occurrence of style k. The field-class conditional field-feature density is (substituting above in (1))

$$p(\mathbf{x}|\mathbf{c}) = \prod_{l=1}^{L} \sum_{k=1}^{K} \alpha_k p(\boldsymbol{x}_l | k, c_l)$$

While the above formula accounts for the presence of styles of patterns, it does not model the consistency of style within a field. Thus different patterns in a field are free to derive from different styles. The salient feature of our style-consistency model is that although field-features have mixture distributions induced by styles, within a field all patterns come from the same style.

$$p(\mathbf{x}|\mathbf{c}) = \sum_{k=1}^{K} P[k|\mathbf{c}]\, p(\mathbf{x}|k, \mathbf{c}) = \sum_{k=1}^{K} \alpha_k\, p(\mathbf{x}|k, \mathbf{c})$$

The assumption is that the style of rendering a field is independent of the identity of the field being rendered ($P[k|\mathbf{c}] = P[k] = \alpha_k$).[1]Within each style we then apply the simplifying assumptions as in the singlet model to obtain our model of style-consistent class-conditional field-feature probability.

$$p(\mathbf{x}|\mathbf{c}) = \sum_{k=1}^{K} \alpha_k \prod_{l=1}^{L} p(\boldsymbol{x}_l | k, c_l) \tag{2}$$

Equation (2) forms, for us, the basis of style-consistency modeling, and can be applied to different kinds of distributions, discrete or continuous. In our implementation and experiments we have used mixtures of Gaussian distributions.

For any style k, and pattern-class c, the pattern-feature

[1]For simplicity of notation we have omitted the random variables in probability terms throughout. This should not perpetuate ambiguity, since we use different notations for the "values" of the random variables. Thus $P[c]$ is the probability of class c while $P[k]$ is the probability of style k.

probability is a mixture distribution.

$$p(\boldsymbol{x}|k,c) = \sum_{j=1}^{J(c,k)} \pi_j(c,k) p_{c,j,k}(\boldsymbol{x};\theta_j(c,k)) \qquad (3)$$

where $J(c,k)$ is the number of mixture components (*variants*) in the distribution for class c, style k. $p_{c,j,k}(\boldsymbol{x};\theta_j(c,k)) \triangleq p(\boldsymbol{x}|\text{class } c, \text{style } k, \text{variant } j)$ is the class, style, and variant conditional feature density function. $\theta_j(c,k)$ denotes one or more parameters of the density function $p_{c,j,k}()$, and $\pi_j(c,k)$ are the mixing parameters.

$$0 \le \pi_j(c,k) \le 1, \qquad \sum_{j=1}^{J(c,k)} \pi_j(c,k) = 1 \qquad \forall c,k$$

In our experiments we use the following models.

Style-bound variant (SBV) model: The class and style conditional distribution is of the form

$$p(\boldsymbol{x}|k,c) = \sum_{j=1}^{J} \pi_j(c,k) p_{c,j,k}(\boldsymbol{x};\theta_j(c,k)) \qquad (4)$$

where the number of variants is fixed (for simplicity) for all classes and styles, $J(c,k) = J$. There are $J \times K$ variant distributions per class, each with a different parameter set, $\theta_j(c,k)$, and weight, $\pi_j(c,k)$.

Style-shared variant (SSV) model: The class and style conditional distribution is written as

$$p(\boldsymbol{x}|k,c) = \sum_{j=1}^{J} \pi_j(c,k) p_{c,j}(\boldsymbol{x};\theta_j(c)) \qquad (5)$$

where $J(c,k) = J$ for simplicity. Here the variant distributions (and their parameters) are not dependent on the style. However, for each class the same set of J variants are weighted differently to obtain the distributions for K styles. Note that for a fixed number of variants the SSV model is more general than the SBV model, albeit at the cost of more parameters.

Singlet (SNGL) model: Singlet modeling is the same as modeling with only one style, with the appropriate number of variants per class.

We shall henceforth confine our discussions to SBV and SNGL models only. Please refer to [Sar00] for experiments and discussions on SSV models. The expanded formulae for the SBV and SNGL models are:

$$p_{\text{SBV}}(\mathbf{x}|\mathbf{c}) = \sum_{k=1}^{K} \alpha_k \prod_{l=1}^{L} \sum_{j=1}^{J} \pi_j(c_l,k)\ p_{c_l,j,k}(\boldsymbol{x}_l;\theta_j(c_l,k)) \qquad (6)$$

$$p_{\text{SNGL}}(\mathbf{x}|\mathbf{c}) = \prod_{l=1}^{L} \sum_{j=1}^{J} \pi_j(c_l)\ \ p_{c_l,j}(\boldsymbol{x}_l;\theta_j(c_l)) \qquad (7)$$

Table 1. Comparison of SBV and SNGL models

		SBV(K,J)	SNGL(KJ)
	Variants per class	KJ	KJ
	Style probabilities α	$K-1$	0
	Variant weights π	$KJ-K$	$KJ-1$
K=2	Variants per class	2	2
J=1	Style probabilities α	1	0
	Variant weights π	0	1
K=2	Variants per class	4	4
J=2	Style probabilities α	1	0
	Variant weights π	2	3

In Table 1 we use the abbreviation SBV(K,J) to denote an SBV model with K styles, and J variants per class per style ($K \times J$ variants per class). SNGL($K \times J$) denotes a singlet mixture model with $K \times J$ variants per class. Since different models use different numbers of parameters, they are not directly comparable. In practice, since the variant distributions account for most of the parameters, we can compare different models that have the same number of variant distributions. When we set $J = 1$, the style-consistency model SBV(K,1) and singlet model SNGL(K) have exactly the same number of variant distributions per class, and the same number of parameters. Our experimental design for printed digit recognition is based on this observation.

3 Style-conscious classifiers

A maximum likelihood (ML) field-classifier is a function that maps an input field-feature vector, $\mathbf{x}$, to a field-class, $\Psi(\mathbf{x})$, according to the formula:

$$\Psi_{\text{ML,field}}(\mathbf{x}) \triangleq \arg\max_{(c_1,\ldots,c_L)} p(\mathbf{x}|\mathbf{c}) \qquad (8)$$

Inserting the singlet model (1) into (8) we obtain the formula for the ML singlet classifier (9).

$$\begin{aligned}\Psi_{\text{ML,SNGL,field}}(\mathbf{x}) &= \arg\max_{(c_1,\ldots,c_L)} \prod_{l=1}^{L} p(\boldsymbol{x}_l|c_l) \\ &= \left(\arg\max_{c_1} p(\boldsymbol{x}_1|c_1), \ldots, \arg\max_{c_L} p(\boldsymbol{x}_L|c_L)\right) \\ &= (\Psi_{\text{ML,pattern}}(\boldsymbol{x}_1), \ldots, \Psi_{\text{ML,pattern}}(\boldsymbol{x}_L)) \qquad (9)\end{aligned}$$

The simplification was obtained by observing that to maximize $p(\mathbf{x}|\mathbf{c})$, a product of L terms with no shared variables, we can maximize each term independently. The process is thus equivalent to ML classification of the patterns, one at a time, and juxtaposition of the assigned pattern-classes to obtain a field-class. This is why we call this model the singlet model.

A maximum likelihood style-conscious classifier is obtained by inserting a style-consistency model for the field-feature probability.

$$\Psi_{\text{ML,SBV,LO}}(\mathbf{x}) = \arg \max_{(c_1,\dots,c_L)} \sum_{k=1}^{K} \alpha_k \prod_{l=1}^{L} \sum_{j=1}^{J} \pi_j(c_l,k)\ p_{c_l,j,k}(\boldsymbol{x}_l;\theta_j(c_l,k)) \tag{10}$$

We call the above classifier a *label only* (LO) or *top-label* classifier since it identifies the top (most probable) label of the field. This is to distinguish it from the following suboptimal approximation which identifies the top field-label and style.

A suboptimal approximation: For a field of length L there are C^L competing field labels for each of which the likelihood function has to be computed for comparison in (10). Thus the computational cost of classification with label-only classifiers grows exponentially with the length of the field. For a long field, assuming that in (10) the term corresponding to the true style of the field $k = k^*$ will dominate the outermost summation over all styles, we can replace this summation by a *maximum*. This approximation is, of course, sub-optimal, but leads to a substantial reduction in computation. It is equivalent to running K style-specific pattern-classifiers, and choosing the output of the one that yields maximum field-feature likelihood (weighted by the a priori style probability α_k). We call such a classifier a top *label-style* (LS) classifier, since it picks out the top field-label and style.

$$\Psi_{\text{ML,SBV,LS}}(\mathbf{x}) = (c_1^{k^*},\dots,c_L^{k^*}) \text{ where} \tag{11}$$

$$k^* = \arg \max_{k=1\dots K} \alpha_k \cdot \prod_{l=1}^{L} \sum_{j=1}^{J} \pi_j(c_l^k,k)\ p_{c_l^k,j,k}(\boldsymbol{x}_l;\theta_j(c_l^k,k))$$

$$c_l^k = \arg \max_{c_l=1\dots C} \sum_{j=1}^{J} \pi_j(c_l,k)\ \ p_{c_l,j,k}(\boldsymbol{x}_l;\theta_j(c_l,k))$$

Note that all maximum likelihood field classifiers are also easily transformed to the respective maximum a posteriori (MAP) classifiers since the field-class probability, $p(\mathbf{c}) = p(c_1,\dots,c_L)$, is provided by a model (such as a linguistic model) that is assumed to be independent of the style of rendition of the field.

$$\Psi_{\text{MAP,field}}(\mathbf{x}) \triangleq \arg \max_{(c_1,\dots,c_L)} p(\mathbf{x}|\mathbf{c}) \cdot p(\mathbf{c}) \tag{12}$$

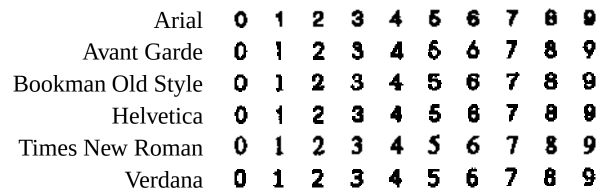


Figure 2. Examples of bitmaps of the 10 digits from 6 different fonts

4 Estimation of model parameters

If patterns in a training set are labeled by class and style, the style consistency model can be trained by partitioning the sample set by class and style. Parameters of class-style conditional distributions and mixing parameters can be estimated from samples of the corresponding partition (may require an iterative method for the mixture distributions). Style-probabilities can be estimated by computing the relative frequencies of patterns from each style.

However, training with style-labeled patterns (*style-supervised training*) is often impractical. Training samples are often not labeled by style. For patterns such as handwritten characters styles may not be well defined (it is difficult to partition 1000 writers into 4 styles). Even when style labels are available (e.g., font-labels for printed text) it may be necessary to model fewer styles than present in order to obtain good parameter estimates from a finite sample. Last, but not least, pre-assigned style labels (such as font-labels) may have little bearing on the feature-measurements used for classification. We model style-consistency in pattern measurements. *Style-unsupervised training* of our model, such as maximum likelihood estimation (MLE) via an Expectation-Maximization (EM) algorithm [Sar00], allows us to characterize styles in a way that bears on the observed distribution of pattern-features.

5 Experiments on machine printed digits

We designed laboratory experiments to demonstrate the application of style-consistency modeling to Optical Character Recognition (OCR). Data were obtained by printing the digits 0 - 9 in six different fonts at 6-point size, on a 600 dpi Apple LaserWriterSelect, and scanning them into bilevel images at 200 dpi on a HP flatbed scanner. Samples of digit bitmaps are shown in Figure 2.

The pattern-feature vectors comprised four central moments, M_{00}, M_{20}, M_{02}, M_{11}, computed for each bitmap.

$$M_{mn} \triangleq \sum_{x=1}^{W} \sum_{y=1}^{H} b(x,y)(x-x_0)^m(y-y_0)^n \tag{13}$$

Table 2. Error rates (%) of different classifiers for printed digit data

Best of 6 mono-font classifiers	43.5
Multi-font singlet classifier (1 Gaussian per class)	34.2
Multi-font singlet classifier (6 Gaussians per class)	19.8
Multi-font style conscious classifier (fields of length 2)	16.5
Multi-font style conscious classifier (fields of length 4)	14.9
Font-specific classifier (1 Gaussian per class per font)	14.2

where $b(x, y)$ is 1 if the pixel at column x, row y of the digit bitmap is black, and 0 otherwise. W and H are the width and height of the bitmap in pixels. (x_0, y_0) is the centroid of the bitmap, *i.e.*, $M_{10} = M_{01} = 0$.

For these experiments, we have tried to keep the generation process for pattern samples (printer, paper quality, scanner parameters) unchanged, so that we can expect that different styles derive only from fonts. For training, 14430 digit patterns (all fonts and equally represented) were arranged in iso-font fields of length 13. The field-classes cycled through the digits (0123456789012, 3456789012345, ...). 2500 new patterns per font (all classes equally represented) were then randomly permuted and partitioned into iso-font fields of length L to obtain test-samples. Two test-sets were created corresponding to L=2 and L=4.

The following classifiers were trained:

- *6 mono-font classifiers*: Singlet classifiers with 1 Gaussian variant per digit-class were trained for each font, from font-labeled digit samples. [10 variants per classifier].

- *Multi-font singlet classifier (1 Gaussian per class)*: [10 variants].

- *Multi-font singlet classifier (6 Gaussians per class)*: A singlet classifier with 6 variants per class, allowing for the possibility of a separate variant for each font: [6×10 variants].

- *Multi-font style conscious classifier*: A 6-style SBV classifier with 1 Gaussian per class-style, obtained by style-unsupervised training (EM algorithm [Sar00]): [6×10 variants].

Note that while the variant distributions in a model account for most of the parameters, the last two classifiers have exactly the same number of parameters (including style-probabilities and mixing parameters – see Table 1). Thus it is fair to compare the classification accuracy of the two classifiers to demonstrate the benefit of style-consistency modeling.

Table 2 lists the percentage of test-patterns misclassified by each classifier. Each of the six mono-font singlet classifiers was applied separately to the entire test set, and the best of them was in error on 43.5% of the samples. The multi-font singlet classifier with the same number of variants, but trained on all fonts, generalizes better and has an error rate of 34.2%. On allowing six variants per class to model the six styles, a singlet classifier can perform much better (19.8%) indicating the multi-modal nature of the class-conditional distributions.

However, font-specific classifiers perform even better on the average (14.2%). This benchmark for style-conscious classification was obtained by applying, to each test field, the specific mono-font classifier corresponding to the font-label of the field. In practice, however, the style-label of a field is not available to the classifier. Our style-consistent top-label (LO) classifier lowers the error rates from the singlet rate (19.8%) towards the benchmark (14.2%). When classifying fields of length two and four, the error rates are 16.5% and 14.9% respectively. Longer fields favor style-conscious classification because more patterns can furnish more information about the style of the field.

6 Discussion and conclusion

We have presented the motivation, a terminology, and a mathematical formulation for style-consistency modeling. Consistency of style is represented in our hierarchical mixture model for the field-class conditional field-feature probability; each field of patterns is rendered in a single style, and within a field each pattern-feature is generated independently according to a class-and-style conditional distribution. The parameters of the model can be estimated from fields of iso-style patterns even if they are not labeled by style.

It has been known for long in pattern recognition communities that style-specific classifiers yield higher classification accuracy than multi-style classifiers. Methods have been presented to identify fonts of documents [ZI98], and to classify handwriting styles based on global characteristics [BVSE97]. Tenenbaum and Freeman suggest a method for factorizing patterns into style and content [TF97]. We do not attempt to identify features that describe styles, but concentrate on the effect of styles on field-feature probability distributions. A suboptimal style model is presented in [BSM99].

Others have demonstrated methods and foundations for adapting classifiers to sufficiently long fields of characters [NJ66, BN94, XN99, Cas86, Bre01]. The advantage of adaptation lies in the potential ability to generalize to a previously unseen style. The strength of our method lies in unsupervised training to a large number of styles and potential application to possibly short style-consistent fields such as hand-written entries in forms, or speech-segments in directory assistance calls.

Optimal style conscious classification is exponential in the length of the field and is impracticable for long fields. The sub-optimal approximation is however only K times as expensive as singlet classification, where K is the number of styles in the model. The above comments on time-complexity apply only to the number of comparisons for the respective maximization steps. Feature-measurement for each pattern, and computation of variant conditional probability density functions induce identical computational overheads for style-conscious and singlet classifiers that have the same number of variants.

In our experiments on machine-printed digit classification, style-consistency modeling reduces classification errors by nearly 25%. While error rates presented here do not represent the state of the art in digit recognition (we use only simple features - the first four central moments, and our Gaussian variants admit only diagonal covariance matrices), our experiments do demonstrate the benefit of style-consistency modeling to pattern classification, and home-grown data allow us to compute and compare the performance of other "comparable" models.

In other experiments conducted on hand-written digit patterns, relative improvement due to style-consistency modeling held up at lower error rates (errors reduced from 5.5% to 4.6%) [Sar00]. Our mathematical framework applies readily to continuous and discrete valued features, where variants are represented by probability density functions and probability mass functions respectively. Style-consistency modeling may also be useful in other applications such as compression.

Acknowledgment.
We thank Dr. H. Fujisawa and his group, at the Hitachi Central Research Laboratories, Japan, for suggesting and supporting this project. This work has been conducted in the New York State Center for Automation Technologies (CAT), which is partially supported by a block grant from the New York State Science and Technology Foundation.

References

[BN94] H. S. Baird and G. Nagy. A self-correcting 100-font classifier. In L. Vincent and T. Pavlidis, editors, *Document Recognition, Proceedings of the SPIE*, volume 2181, pages 106–115, 1994.

[Bre01] T. Breuel. Modeling the sample distribution for clustering OCR. In *Document Retrieval VIII, Proceedings of SPIE*, volume 4307, 2001.

[BSM99] I. Bazzi, R. Schwartz, and J. Makhoul. An omnifont open-vocabulary OCR system for English and Arabic. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-21(6):495–504, June 1999.

[BVSE97] V. Bouletreau, N. Vincent, R. Sabourin, and H. Emptoz. Synthetic parameters for handwriting classification. In *Proceedings of the Fourth International Conference on Document Analysis and Recognition*, volume 1, pages 102–106, Ulm, Germany, 1997.

[Cas86] R. G. Casey. Text OCR by solving a cryptogram. In *Proceedings of the Eighth ICPR*, pages 349–351, Paris, 1986. IEEE Computer Society Press.

[Jel97] F. Jelinek. *Statistical methods in speech recognition.* The MIT Press, Cambridge, MA, 1997.

[NJ66] G. Nagy and G. L. Shelton Jr. Self-corrective character recognition system. *IEEE Transactions on Information Theory*, IT-12(2):215–222, April 1966.

[Sar00] P. Sarkar. *Style consistency in pattern fields.* PhD thesis, Rensselaer Polytechnic Institute, Troy, NY, USA, 2000.

[SN00] P. Sarkar and G. Nagy. Classification of style-constrained pattern-fields. In *Proceedings of the fifteenth ICPR*, pages 859–862, Barcelona, 2000. IEEE Computer Society Press.

[TF97] J. B. Tenenbaum and W. T. Freeman. Separating style and content. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems 9*. Morgan Kaufmann, San Mateo, CA, 1997.

[XN99] Y. Xu and G. Nagy. Prototype extraction and adaptive OCR. *IEEE Transactions on PAMI*, 21(12):1280–1296, 1999.

[ZI98] A. Zramdini and R. Ingold. Optical Font Recognition using typograpgical features. *IEEE Transactions on PAMI*, 20(8), August 1998.