CLASSIFICATION OF DIGITAL TYPEFACES USING SPECTRAL SIGNATURES

ROBERT A. MORRIS

Department of Mathematics and Computer Science, University of Massachusetts at Boston, Boston, MA 02125-3393, U.S.A.

Abstract—Mean Fourier amplitudes through a bank of bandpass filters provide a feature vector with which typefaces can be identified using a piecewise quadratic classifier.

Fourier amplitudes   Typeface classification   Quadratic classifier   Digital type   Digital fonts

1. INTRODUCTION

In this paper we describe the results of classification experiments we have performed on text images in various digital fonts. The feature vectors in the study are derived from data in the Fourier amplitude spectra of the images. These experiments have several purposes: (a) to establish statistical verification of certain predictions made by image processing theory; (b) to provide a foundation for automatic typeface identification of OCR data; (c) to examine the applicability of human vision models to the identification and discrimination of typefaces and (d) to investigate whether spectral features might be useful in typeface production, an enterprise which has important algorithmic components for which no quantitative measures of success presently exist.

Nowadays, most documents are produced with characters made—at least initially—in digital form under computer supervision. Laser printers and photo-typesetters make their images by selecting which pixels to darken. (The latter only 20 years ago rendered their characters photographically. For a review of the technology and history of digital type rendering, see for example reference (1).) The final appearance of a character on a page depends on many factors, central among which is the scan conversion—the choice of which pixels to darken. Sometimes this is done by extensive bitmap editing by trained designers, especially for low-to-medium resolution devices such as 300 dots per inch (dpi) laser printers. In recent years, the scan conversion has increasingly been done algorithmically,[2,3] often on demand in the printing device itself. This trend is increasing because of the advantages in cost, time, and flexibility of algorithmic scan conversion.

Many gross distinctions between typefaces are easily made by the untrained observer, e.g. serif vs. sanserif faces and variable-width designs vs. fixed-width ones. Other distinctions, while noticeable, are not so easily characterized. The impact on the reader of different typefaces and sizes in a document arises from non-trivial skills usually residing in the mind of the typo-grapher. These professionals have various informal classification schemes for their choices—some traditional and widely used, others personal and unarticulated. Attempts at such classifications and characterizations by these professionals and those who design the letterforms are the subject of substantial critical writing, especially of individual typeface designs,[4,5] but there have been only the simplest statistical studies of any identifying or classifying properties of these images, e.g. reference (6), which contains histograms showing the distributions for several typographic variables on a sample of over 700 typefaces, together with some classification terminology based on these variables.

2. SPECTRA OF TEXT IMAGES

We first wish to explain why the statistics of the Fourier amplitude spectra might be expected to provide insight into the issues mentioned above. An image i may be regarded as a real-valued function on the plane, where $i(x, y)$ denotes the image intensity at (x, y). In practice, we consider images which are only of finite extent (this leads to certain well-known technical issues in computing their spectra). The *Fourier transform* $I(\omega_x, \omega_y)$ of $i(x, y)$ is given by

$$I(\omega_x, \omega_y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} i(x, y) e^{-j2\pi(x\omega_x + y\omega_y)} \, dx \, dy.$$

It can be written in polar coordinates as

$$A(\omega_x, \omega_y) e^{j\phi(\omega_x, \omega_y)}$$

where $A(\omega_x, \omega_y) = |I(\omega_x, \omega_y)|$ is the *Fourier amplitude* at (ω_x, ω_y) and $\phi(\omega_x, \omega_y)$ the *Fourier phase* at (ω_x, ω_y).

The variables ω_x and ω_y denote *spatial frequency* and (roughly) represent the rate at which the image is changing in its intensity in the x (respectively y) direction. High spatial frequencies correspond to rapid rates of change with respect to position, i.e. of small features in the image. Low spatial frequencies correspond to slow rates of change, i.e. of large features.

Amplitudes and phases have fairly intuitive interpretations. The amplitudes denote the amount of variation at the given spatial frequency. Large amplitudes at a given spatial frequency or band of frequencies accompany prominent features repeating at those frequencies. For example, in an image of text set in a well designed typeface, the vertical strokes in the characters are regularly spaced and will therefore contribute a relatively large amplitude at a frequency $(\omega_x, 0)$. We might call this the *stroke frequency*. (In a single line of type, the strokes are only evenly spaced horizontally, so they have no variation in the y direction.) On the other hand, in an *ensemble* of images in a fixed type size, we might expect this amplitude to have little variance and so contribute little to any classification based on amplitudes. (It is not completely correct that stroke spacing is constant: type designers do sometimes change the spacing when making variants of a typeface, but by and large the stroke frequency is more characteristic of the size than of anything else.)

Thus amplitude encodes the local size variations in features. Phases, on the other hand, encode the gross position of features. More precisely, if an image i is translated by a vector (x_0, y_0) then the new image has the same amplitude but has linearly shifted phase

$$\phi'(\omega_x, \omega_y) = \phi(\omega_x, \omega_y) + x_0\omega_x + y_0\omega_y.$$

For these reasons, it is natural to suspect that the amplitude might have typeface-specific (and character-independent) information, whereas the phase might have character-specific (and typeface-independent) information. In fact, this role of phase seems to be confirmed.[7]

Now we can describe the feature vector we use to classify such text images. Given a text image, our problem is to identify the typeface to which the characters of that image belong. We have made several important simplifying assumptions, the relaxation of which is interesting and is the subject of our further research.

First, our images are noise-free. That is, we are not deriving them from scanned images of paper documents, but rather synthetically from bitmaps for the characters in each typeface. (We remark below on the nature of the bitmap generation and what interest there is in studying *its* effect.) The application of these techniques to OCR data is potentially important because the typeface in which a string is rendered can help in understanding the structure of a document (for example, italics sometimes lend emphasis, computer programs are often set in fixed-width fonts, etc.). Automatic structure recognition is an active area of research.[8,9] Although it is reasonable to imagine that the noise statistics will be constant across typefaces, we have not yet attempted to extend our experiments to OCR data. Of course, the independence of noise across typefaces is *not* true when certain aspects of the image rendering are considered. For example, it has been well known for many years that the precise shape of letter features can affect the way in which ink is deposited, especially

in high-speed, low-quality applications such as newspaper printing. Small-angled joins in characters can cause so-called "ink traps" which fill with ink and distort the feature. These considerations do apply to laser printing also, but their precise effect has only recently begun to be explored.

Second, we have used only two specific pieces of software to produce the bitmaps. These are Metafont, a public domain font production system written by Donald Knuth[10] and TypeScaler, a commercial font production system from Sun Microsystems.[11] Many more rendering systems are in use, and it will be interesting to see whether the techniques of this paper can, on the one hand, discriminate between renderings of the same face by different software, and, on the other hand, correctly identify a given face no matter what the rendering system.

Third and last, in the experiments reported here we have restricted our attention to 10-point type rendered at 300 dpi for Canon CX laser printer marking engines, which were the most commonly used by printer manufacturers until recently replaced with Canon's SX engine. Each of the rendering programs can take account of (some) physical properties of the marking engine, and it will be important to see if *those* choices reflect themselves in our classifications.

3. THE FEATURE SPACES IN THE STUDIES

The samples under study consist of strings selected at random from English text (a portion of *Wuthering Heights* for training our classifier and a computer science grant proposal for testing it). Each string is rendered in a particular typeface as described above and truncated to an image 512 pixels wide and 64 pixels high, then passed through a Blackman filter to control artifacts arising from the truncation (see p. 447 of reference (12)). Then 512×64 discrete Fourier transform of the string is computed and the amplitudes at each of the resulting points recorded. Since 10-point type at 300 dpi is typically about 30 pixels high, our sampling has the effect of embedding every text string in a box with approximately constant white space surrounding the string. We assume this white space has no effect on the statistics within each class.

To motivate our construction of the statistical features, we observe that the graphical features of text are of several different scales. For example, serifs are quite small, but bowls are relatively large. It is therefore reasonable not to consider the entire amplitude spectrum at once, but rather to pass it through a bank of bandpass filters, each of which leaves us with data about a particular range of frequencies only. This approach is also reasonable from a visual point of view, since contemporary vision scientists largely agree that the visual system does exactly that in the first stages of its image processing.[13]

Hence, we consider a collection of N linear filters, or, more conveniently, their Fourier transforms F_k regarded as functions on the (ω_x, ω_y) frequency domain.

The result on the spectrum of applying the filters is obtained by multiplying the amplitudes by F_k. Since we are not interested in phases, we can restrict our attention to filters which have no effect on phase (i.e. which do not shift the images), as is true of filters currently used to model human vision.

We considered two families of filters. First is a collection of ideal filters, which by definition are the identity on a particular region of the frequency space and zero outside that region. The other family is that of a particular model of human vision, which we describe later. We take as the kth feature the mean Fourier amplitude after applying the kth filter. If the passband of the filter is sufficiently small (thus, we need many filters to cover the 512×64 samples in the frequency domain), this mean amplitude might be a good approximation to the amplitude throughout the passband; this alone might lead to successful classification of the spectra.

In the classification using ideal filters, we first partitioned the spatial frequency plane into 38 rectangular regions as described in Appendix C. The regions have very little overlap, except for those at the highest spatial frequencies. For each region we define a corresponding bandpass filter whose value is 1 on the region and 0 off it. (By principal component analysis we can account for 99% of the variance between classes with 17 filters. It is those 17 filters—which are linear combinations of the original 38—which form the filter set we actually use for the final classification.) Each of the small bandpass filters represents about 4% of the spectrum in each direction, and the three highpass filters begin at 36% of the spectrum. †

Assuming an 18 in. viewing distance, then with the sampling we have described for the construction of the type images, the maximum spatial frequency represented on each axis is about 96 cycles per degree (cpd) of visual angle. (This is the usual measure of spatial frequency in vision research, since it is distance independent.) The features given by the highpass filters thus represent mean amplitude above about 35 cpd, corresponding to features smaller than about 2 min of visual angle. However, most humans cannot distinguish features smaller than 1 min of visual angle (i.e. 60 cpd) and visual acuity for many tasks drops off rapidly above 15 cpd. This suggests that features smaller than 4 min wide will not play much of a role in distinguishing type images from one another. Thus, if we have significant energy at these high frequencies, we can expect that it represents either harmonics of larger features or aliasing artifacts due to truncation of the images. One important exception to this is the serifs present in some typefaces, which we will discuss later. As mentioned above, we control the aliasing artifacts with standard windowing techniques. Harmonics might

be expected to show up in correlations between filters, which could also provide a basis for dimensionality reduction in the classification.‡ In summary, we have partitioned the spectrum finely where there is some reason to believe features are visually relevant, and coarsely elsewhere.

The second family of filters is that described by the model of human vision put forth by Wilson and Gelb.[14] These are linear combinations of Gaussian functions with coefficients chosen to fit experimental data. Their precise form is not relevant to this discussion, but we should note two things. First, unlike our ideal filters, each of these filters overlaps substantially with its nearest neighbors (measured either by their separation in the x or y direction, or by their orientation tuning). This means that averaging the filtered images generally counts each amplitude two or more times. In addition, there is some suggestion in the vision literature that these models cannot account for pattern classification (p. 434 of reference (15)). On the other hand, vision filtered features have found some success in texture recognition algorithms.[16,17] Second, the vision filters we used have much greater bandwidths than the ideal filters we used, and for this reason averaging the response through these filters may be throwing out useful classifying data. In general, we got poor results with these filters and have therefore abandoned them. One experiment with ideal filters of comparable bandwidths similar to those of the vision filters led to classification rates intermediate between those with the vision filters and those we report here.

We assumed that the class conditional densities are normal and used a piecewise quadratic classifier as an approximation to the Bayes minimum error classifier (p. 169 of reference (18)). The class statistics were gathered on 100 text strings. The classification success rate was estimated by classifying 100 strings. We briefly examined histograms of marginal distributions for a few cases and satisfied ourselves of the reasonableness of the normalcy assumption, but made no rigorous attempt to verify that the feature vectors are normally distributed; this assumption is in fact lying under the use of a quadratic classifier as the minimum error classifier. Indeed, we know that certain spectral features of text images are found in all text (of a fixed size). For example, the distance between vertical stroke centers in adjacent 10-point characters (the most common for reading, and the size used for all our experiments) is generally constant at about 7.5 min of visual arc. This means the most 10-point text in western languages has an amplitude peak at $60/7.5 = 8$ cpd. The spectral samples therefore will not be independent in the vicinity of this characteristic frequency. Another common spectral feature arises from the fact that our samples are always 512 pixels wide, which corresponds

---

† Strictly speaking, these are also bandpass filters, since the entire spectrum is truncated by virtue of sampling, but we abuse terminology and refer to them as highpass.

‡ We briefly examined correlation coefficients and found adjacent filters to be highly correlated in general, but we did not explore this as an avenue of dimensionality reduction.

to about 12 characters. Since English word length averages 4–5 characters, about half our samples have one word boundary and half have two. This leads to a bimodal distribution of means through our low frequency filters corresponding to a "word frequency" much akin to the stroke frequency described above. As both of these phenomena are relatively constant across typefaces, we assume that they do not contribute to classification and thus ignore them.

Preliminary experiments with a nearest mean classification gave very poor results. With the current experiments we indeed found that the class covariance matrices vary widely. This suggests that success with a linear classifier is unexpected, since the quadratic classifier reduces to the nearest mean classifier only when the covariances are the same.

4. CLASSIFICATION RESULTS

Table 1 summarizes the results of classification on all 55 fonts in the study. For the ideal filter based features, overall error rate is about 6%. Three fonts were perfectly classified, and half had error rates below 3%. In a second study on a substantially smaller number (nine) of typefaces classification improved somewhat to an overall error rate of about 4% (Table 2).

More realistic experiments focus attention on related collections of typefaces, such as might be found in a single document. Table 3 gives the overall classification rate among the variants in each of four families: Lucida, Avant Garde, Pandora, and Computer Modern.

Table 1. Summary of classification statistic for two experiments classifying 55 fonts. The first column describes the feature space comprising spectral amplitudes through ideal filters, the second column through filters from a model of human vision. Experimental design is described in the text

	Ideal filters	Vision filters
Minimum class error	0	0.03
Maximum class error	0.12	0.47
Median class error	0.03	0.18
Total error	0.06	0.19

Table 3. Classification error rate within four typeface families. N_C is the number of fonts in the study, ε the total error for the family

	Avant Garde	Computer Modern	Lucida	Pandora
N_C	4	8	10	6
ε	0.003	0.006	0.019	0.157

Each entry represents an experiment in which the classifier was trained on 100 samples in each class and tested on 100 samples taken from a different text. N_C is the number of classes (i.e. the number of typeface variants in each family), and ε the overall error rate as estimated by the number of misclassified samples in the testing.

In light of the above classification rates, it is instructive to consider the nature of these families and—in the case of Lucida and Pandora—to examine their confusion matrices. The reader is invited to examine type samples in Appendix A to appreciate some of the remarks below.

The Avant Garde family was originally designed in 1962 by Herb Lubalin as logotype for the magazine of the same name. Perhaps this is why Avant Garde faces are somewhat taller than similar type of the same width. It is sometimes argued that their original design makes them inappropriate for general use in small sizes.[19]

The Computer Modern family[20,21] was designed by a computer scientist (Donald Knuth) with the assistance of professional type designers in later stages of the design. It is a "loosely coupled" family in that only some of its variants are closely related visually. For example, the serif and sanserif faces are quite different, unlike, e.g. those of Pandora (see below).

The Lucida family, designed by Charles Bigelow and Kris Holmes, was among the first designed explicitly for low-to-medium resolution digital media. Its designers took special care to solve certain problems endemic to 300 dpi rendering on laser printers.[22] The bright variant, designed especially for high resolution typesetter use, is the main text face in which *Scientific American* is typeset. In the experiments described in

Table 2. Confusion rates for 100 samples from each of nine fonts. The overall correct classification rate is 0.96. Design is 17-dimensional principal components derived from 38 quadratic discriminant functions given by mean spectral amplitude through non-overlapping ideal filters, as described in the text. The components account for 99% of the variance in the original classifier. (Font name abbreviations are listed in Appendix B)

	AvGBk	Bebmbo	Cour	GlSans	LucBri	LucSans	TimesR	Helv	cmr	
AvGBk	0.95							0.05		AvGBk
Bebmbo		0.92					0.03			Bebmbo
Cour			0.98				0.02			Cour
GlSans				0.99		0.01				GlSans
LucBri					0.99	0.01				LucBri
LucSans					0.02	0.98				LucSans
TimesR		0.05					0.95			TimesR
Helv	0.05							0.95		Helv
cmr		0.04					0.01		0.95	cmr

Table 4. Confusion rates for 100 samples from each of 10 members of the Lucida family. Font name abbreviations are given in Appendix B

	LB	LBD	LBDI	LBI	LS	LSB	LSBI	LSI	LST	LSTB
LB	1.0									
LBD		1.0								
LBDI			1.0							
LBI				1.0						
LS	0.01				0.91				0.08	
LSB		0.01				0.98				0.01
LSBI							1.0			
LSI								1.0		
LST					0.10				0.90	
LSTB										1.0

Table 4, the four Lucida Bright styles are distinguished from one another 100% of the time. Virtually the only confusion among Lucida faces is between Lucida Sans and Lucida Sans Typewriter, which we discuss below.

The Pandora typeface family was designed by Neenie Billawala as an exercise in the use of Metafont, rather than for production use. A central design criterion was uniformity of appearance among the members of the family.[23] Our methods somewhat reflect the success in that they failed to cleanly separate all the members of the Pandora family from one another (in contrast to the virtually 100% classification rates for the other families shown in Table 3).

One kind of distinction among typefaces, especially within tightly coupled families such as Pandora, is often expressed in the serifs. Because serifs are small, they are not visible through low frequency filters, but find their expression in the high frequencies. Therefore, we consider briefly exactly what information is contained in the high-frequency energy and what is the cost of ignoring it or of averaging it in the gross way we described above.

The highest spatial frequency which can be represented with sampled images is 1 cycle per sample, i.e. 1 cycle per pixel (cpp). Since our highpass filter has its low edge at 36% of the maximum frequency, it is selecting features which are smaller than 1/0.36 (roughly 3) pixels. This is approximately the size of the serifs in the characters in our experiment. To the extent that we lose information by our treatment of the high frequencies, we can expect increased confusion where the serifs are an important discriminator.

The most striking example of this is in the Pandora family. When we classify only within this family, we get high confusion rates—averaging 16% error (Table 5), compared to, for example, 2.1% error when we classify only within the Lucida family (Table 4), whose serifed and sanserif variants differ in ways other than the presence of serifs. In the Pandora classification the confusion is virtually always between a serif face and its corresponding sanserif face—because by design these variants differ almost entirely in the serifs. The only consequential confusion in the Lucida family is between Lucida Sans and Lucida Sans Typewriter.

Table 5. Confusion rates for 1^^ samples from each of six typefaces from the Pandora family. The overall correct classification rate is only 84%. Most misclassification is between the serifed variants and their corresponding sanserif face, because in Pandora the serifs are the principal visual discriminant between each such pair, and these features are too small to be distinguished without finer filter. However, as described in the text, finer filters in fact suffice to separate the pairs. (Typeface name abbreviations are listed in Appendix B)

	pnr	pnss	pnb	pnssb	pnsl	pnssi	
pnr	0.82	0.16					pnr
pnss	0.19	0.79					pnss
pnb			0.85	0.15			pnb
pnssb			0.11	0.89			pnssb
pnsl					0.85	0.15	pnsl
pnssi					0.16	0.84	pnssi

The latter is a fixed width font highly stylized to look as much like Lucida Sans as possible.[19] We were able to increase the average success rate to 96% with Pandora by doubling highpass cutoff (even while doubling the width of the filters to keep the computation small).

Finally, we note that two of the families (Lucida and Avant Garde) were produced with the TypeScaler software, and two with Metafont (Computer Modern and Pandora), so the font production system alone is not likely to account for the confusions described above. Further, when we put together all the members of each family and run the classifier in an attempt to classify between rather than within families, we get family separations ranging from 92 to 97% correct identification, with an average correct identification of over 96%.

5. SUMMARY

Collections of mean Fourier amplitudes through ideal filters form a reasonable feature space for classifying typefaces because typefaces differ from one another by features of varying scale. Samples of typefaces which are visually similar on these grounds are confused by our spectral measures, while those which are not visually similar are well separated.

REFERENCES

1. R. A. Morris, Rendering digital type: a historical and economic view of technology, *The Comput. J.* **32**(6), 524–532 (1989).
2. R. D. Hersch, Introduction to font rasterization, *On Electronic Publishing, Proc. Int. Conf. on Raster Imaging and Digital Typography*, Ecole Polytechnique Fédérale Lausanne, October 1989, J. André and R. D. Hersch, eds, pp. 1–13. Cambridge University Press, Cambridge (1989).
3. J. D. Hobby, Generating automatically tuned bitmaps from outlines, Technical Report No. 148, AT&T Bell Laboratories, Murray Hill, New Jersey, February (1990).
4. W. Tracy, *Letters of Credit. A View of Type Design.* David R. Godine, Boston (1986). U.K. edition published by Gordon Fraser, London (1986).
5. C. Bigelow, P. H. Duensing and L. Gentry (eds), *Fine Print on Type.* Fine Print, P.O. Box 3394, San Francisco (1989).
6. H. Berthold, *Berthold Fototypes E2*, Vol. 1. H. Berthold AG, Berlin (1980).
7. G. Duan and R. A. Morris, The importance of phase in the spectra of text, *Electron. Publishing* **2**(1), 47–59 (April 1989).
8. L. D. Wilcox and A. L. Spitz, Automatic recognition and representation of documents, *Text Processing and Document Manipulation: Proc. Int. Conf. on Electronic Publishing*, New York, April 1988, J. C. van Vliet, ed., pp. 47–57. Cambridge University Press, Cambridge (1988).
9. R. Ingold, R.-P. Bonvin and G. Coray, Structure recognition of printed documents, *Text Processing and Document Manipulation: Proc. Int. Conf. on Electronic Publishing*, New York, April 1988, J. C. van Vliet, ed., pp. 59–70. Cambridge University Press, Cambridge (1988).
10. D. E. Knuth, *The Metafontbook*, Computers and Typesetting, Vol. C. Addison-Wesley, Reading, Massachusetts (1986).
11. Sun Microsystems Inc., *TypeScaler Reference Manual*, Release 1.3 (1989).
12. A. V. Oppenheim and R. W. Schafer, *Discrete Time Signal Processing.* Prentice-Hall, Englewood Cliffs, New Jersey (1989).
13. H. R. Wilson, D. Levi, L. Maffei, J. Rovamo and R. De Valois, *The Perception of Form: Retina to Striate Cortex*, Chap. 10, pp. 231–272. Academic Press, San Diego (1990).
14. H. R. Wilson and D. J. Gelb, Modified line element theory for spatial frequency and width discrimination, *J. Opt. Soc. Am.* **1**, 124–131 (1984).
15. R. Shapley, T. Caelli, S. Grossberg, M. Morgan and I. Rentschler, *Computational Theories of Visual Perception*, Chap. 15, pp. 417–448. Academic Press, San Diego (1990).
16. T. M. Caelli, W. F. Bischof and Zhi-Qiang Liu, Filter-based models for pattern classification, *Pattern Recognition* **21**, 639–650 (1988).
17. J. M. Coggins and A. K. Jain, A spatial filtering approach to texture analysis, *Pattern Recognition Lett.* **2**, 195–203 (1985).
18. K. Fukanaga, *Introduction to Statistical Pattern Recognition*, 2nd Edn. Academic Press, New York (1990).
19. C. Bigelow, Personal communication (1990).
20. D. E. Knuth, *Computer Modern Typefaces*, Computers and Typesetting, Vol. E. Addison-Wesley, Reading, Massachusetts (1986).
21. D. E. Knuth, Lessons learned from Metafont, *Visible Language* **XIX**(1), 35–54 (Winter 1985).
22. C. Bigelow and K. Holmes, The design of Lucida: an integrated family of types for electronic literacy, *Text Processing and Documentation, Proc. Int. Conf.*, University of Nottingham, J. C. van Vliet ed. Cambridge University Press, Cambridge (1986).
23. N. Billawala, Pandora: an experience with Metafont, *On Electronic Publishing, Proc. Int. Conf. on Raster Imaging and Digital Typography*, Ecole Polytechnique Fédérale Lausanna, October 1989, J. André and R. D. Hersch, eds. Cambridge University Press, Cambridge (1989).
24. J. André and R. D. Hersch (eds), Raster imaging and digital typography, *On Electronic Publishing, Proc. Int. Conf. on Raster Imaging and Digital Typography*, Ecole Polytechnique Fédérale Lausanne, October 1989. Cambridge University Press, Cambridge (1989).
25. J. C. van Vliet, ed., *Text Processing and Document Manipulation: Proc. Int. Conf. on Electronic Publishing*, New York, April 1988. Cambridge University Press, Cambridge (1988).

APPENDIX A. TYPEFACE SAMPLES

AvantGarde-Book

With that concluding word,

Bookman-Light

With that concluding word,

cmss10

With that concluding word,

Courier

With that concluding word,

Helvetica

With that concluding word,

LucidaSans

With that concluding word,

pnr10

With that concluding word,

pnsl10

With that concluding word,

pnb10

With that concluding word,

Bembo

With that concluding word,

cmr10

With that concluding word,

cmtt10

With that concluding word,

GillSans

With that concluding word,

LucidaBright

With that concluding word,

LucidaSans-Typewriter

With that concluding word,

pnss10

With that concluding word,

pnssi10

With that concluding word,

pnssb10

With that concluding word,

Times-Roman

With that concluding word,

**APPENDIX B. TYPEFACE ABBREVIATIONS IN
TABLES 2, 4 AND 5**

AvGBk	Avant Garde Book
cmr	Computer Modern Roman
Cour	Courier
GlSans	Gill Sans
Helv	Helvetica
LB, LucBri	Lucida Bright
LBD	Lucida Bold Demi
LBDI	Lucida Bold Demi Italic
LBI	Lucida Bold Italic
LS, LucSans	Lucida Sans
LSB	Lucida Sans Bold
LSI	Lucida Sans Italic
LSBI	Lucida Sans Bold Italic
LST	Lucida Sans Typewriter
LSTB	Lucida Sans Typewriter Bold
pnr	Pandora Roman
pnss	Pandora Sans Serif
pnb	Pandora Bold
pnssb	Pandora Sans Serif Bold
pnsl	Pandora Slanted
pnssi	Pandora Sans Serif Italic
TimesR	Times Roman

APPENDIX C. IDEAL FILTERS

The basic bank of ideal filters divides the ω_x and ω_y axes into N intervals $(u_i, u_{i+1}]$ and their reflections around the ω_x-axis, here $u_1 = 0, u_{i+1} - u_i = w$, for $i < N$ together with $(u_N, u_{max}]$. This partitions the frequency plane into overlapping rectangular regions as shown in Fig. C1. In most of the experiments described here, $N = 11$, $w = 0.04$ and $u_N = 0.36u_{max}$. For any sampling, $u_{max} = 1$ cycle per pixel, but as described in the text, this corresponds to about 94 cycles per degree of visual angle for the images we studied if considered to be viewed at an 18 in. distance. The overlaps are quite small for all but the highest band. As discussed in the text, this is generally harmless except for the detection of serifs. For Pandora, the single (and unusual) family where the sanserif and serif variants differ *principally* in the serifs, it is necessary to take a higher frequency cutoff to separate the two variants in the classification. In this case, we used $u_N = 0.80u_{max}$ and increased w to 0.08. In this case, we generally exceeded 96% correct classification for the difficult Pandora faces.
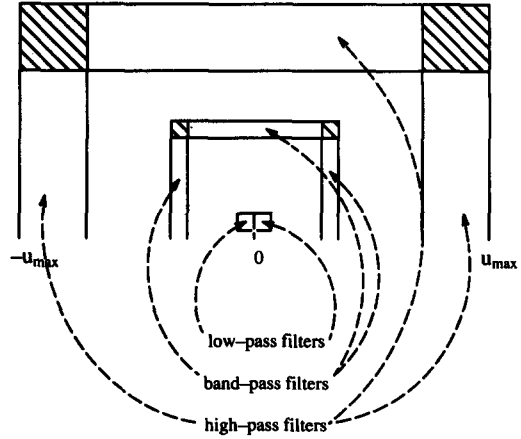

Fig. C1. Support regions of ideal filters. Shaded regions denote overlap.

About the Author—ROBERT MORRIS is Professor of Mathematics and Computer Science at the University of Massachusetts at Boston. He received the B.A. in 1965 from Reed College, and the M.A. (1967) and Ph.D. (1970) in mathematics from Cornell University. His research interests include document processing, image processing aspects of digital typography, and human vision. He was Chair of RIDT91, the Second International Workshop in Raster Imaging and Digital Typography, held in Boston in October 1991, and is a member of the editorial board of the journal *Electronic Publishing, Origination, Dissemination, and Design*.