ELECTRONIC PUBLISHING, VOL. 5(2), 79–89 (JUNE 1992)

Teaching digital typography[1]

JACQUES ANDRÉ
Didot Project
Irisa/Inria–Rennes, Campus de Beaulieu
F-35042 Rennes cedex, France

ROGER D. HERSCH
Didot Project
Laboratoire des Systèmes Périphériques
École Polytechnique Fédérale de Lausanne
CH-1015 Lausanne, Switzerland

SUMMARY

Digital typography is a very specialized field that offers two widely different yet complementary aspects: art and computer science. This paper presents Project Didot, which is all about teaching digital typography. While taking into account recent experience, the authors explore some subjects that should be included in a digital typography course and describe the various trades it would be aimed at. This paper concentrates on the computer science aspect and gives a basic bibliography.

KEY WORDS Digital typography Curriculum Tuition

1 PROJECT DIDOT

In 1990, the EEC launched its Comett II project, with its main aims being to place greater emphasis on advanced technology training and to ensure that cooperation between universities and the industrial world is carried out at a European level.

Project Didot[2] was set in motion in this context, with the help of seven other partners.[3] The aim of this three-year project is mainly to draw up a European curriculum for teaching digital typography,[4] to implement the required software tools and to try out this curriculum in a teaching environment [4].

Among the experimental workshops organized for this purpose [5] was a two-day seminar which took place in Reading (UK) in February 1991 [6] as well as a one-week seminar organized in Lausanne (Switzerland) in September 1991 [7,8]. Whereas the first seminar mainly attracted graphic art professionals, the second one was attended by as

[1] This study is partly financed by the EEC/Comett II project no 90/1/3697/Cb. A first version of this paper appeared in French as "Enseigner la typographie numérique", in J. André (ed.), Proceedings of TEP'92, *Bigre*, 79, March 1992. Opinions stated in this paper are the authors' own and do not necessarily reflect those of the project as a whole.

[2] Didot means DIgitizing and Designing Of Type or DIdacticiels de Dessin par Ordinateur de caractères Typographiques. The name refers to the Didot dynasty and especially to François-Ambroise Didot (1730–1804) who gave us the first typographic unit, i.e., the didot point.

[3] The nine partners in this project include two teaching institutions: the Department of Typography of Reading University (United Kingdom) and the AGS Schule für Gestaltung Art school (Basel, Switzerland); three research institutes: the Peripheral Systems Laboratory of the Swiss Federal Institute of Lausanne (Lausanne, Switzerland), Irisa/Inria-Rennes (France) and EuroPARC (Rank Xerox, Cambridge, United Kingdom) as well as four private companies: URW (Hamburg, Germany), Lg&A-Hito (Getxo-Vizcaya, Spain), Infoprint (Salonica, Greece), and P. de Macchi (Turin, Italy).

[4] As far as we know, there is at present no curriculum for digital typography even though curricula have been drawn up for other computer sciences, for instance the *Curriculum for Computer Education for Management* [1,2] of the ACM (Association for Computing Machinery, USA) and, more recently, those for computer engineering [3].

CCC 0894–3982/92/020079–11

Received 15 July 1992
Revised 31 July 1992

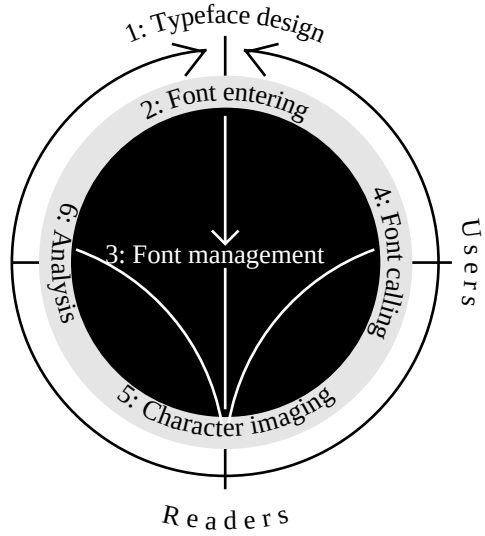

Figure 1. The various processes related to typefaces. Black indicates the computer scientists' world, white shows the typographers' domain, and grey a mixed field. Numbers refer to the subsections in Section 2

many computer scientists as graphic art specialists and resulted in fruitful exchanges. In fact, the Lausanne school included two types of courses (notwithstanding practical work on computers):

- courses in letterpress typography aimed at all participants and especially a series of courses and exercises,[5] *Rhythms, forms, expression,* already tested in Basel [10] and in Reading,
- courses in digital typography mainly attended by computer scientists, which will be further discussed in section 4.

In 1992, two seminars were held in Paris (one on digitizing fonts through Ikarus and one on ligatures), in Basel (more dedicated on the educational approaches to the design and production of digital type) and one in Rennes (France) for teachers. Other courses and seminars are currently being prepared (Hamburg, August 1992, Milan and Thessaloniki in October 1992, etc.). The project publishes a magazine, *Didot bulletin*,[6] where large reports are available on these seminars as well on other activities such as a study on a European vocabulary for digital typography.

2 DIGITAL TYPOGRAPHY: A COMPLEX FIELD

In this paper, the expression *digital typography* does not encompass such a large meaning as the one it covers for some authors [13]; in fact, we are only focusing on methods for creating and drawing characters (in the same way that [14], [15] or [16] deal with the letterpress approach) and not at the higher level of page layout.

Digital typography is a field that overlaps two others: that of classical or letterpress

[5] Including a successful art studio atmosphere, which was lacking in a previous experiment [9].
[6] [11,12]. Copies on request to the authors of the present paper.

typography and that of computer science (Figure 1). Our basic postulate is that digital typography should not be taught without teaching classical typography at the same time. However, this paper does not describe which part of classical typography should be included in a course on digital typography.[7] Our aim is to describe the part related to computer science and mathematics.

In the present state of knowledge (which varies with the emergence of new technology), computerized fonts can be perceived differently, according to the qualifications and the job held by the people who deal with them. Here are a few examples of activities related to digital typography; they imply having different points of view and use solutions which require very different utilities and software.

- For instance, a font designer's task is to create a new font family. For this purpose, he will use either conventional or computer-aided drawing techniques, or even a mixture of both.
- On the other hand, a printing house will require a sophisticated computer system to handle a large number of fonts which are downloaded onto phototypesetters when needed.
- Software companies developing multi-font interactive applications must also find a way of handling available fonts (font name, size, style) and come up with solutions for imitating the rendition of unavailable fonts.

2.1 Creating digitized characters

Font designers often do their drawings by hand and then digitize the outlines manually point by point using an appropriate software such as *Ikarus*. Another way of doing things is to scan the drawing of the font in order to get a bitmap which can then be processed by an automatic outlining software such as *Linus* or *Typo*. The resulting outlines will have to be touched up using an interactive outline editor (*Fontographer*, *FontStudio*, *Typo*), since the rendition of available automatic outline extraction programs is only partly satisfactory. See sections 4.8 and 4.9 below.

Designers interested in the mathematical specification of pen trajectories and shapes can simulate hand-drawn designs by using an appropriate program [18]. This program, METAFONT, is capable of generating families of designs by varying pen size, pen orientation and other shape-creation parameters.

2.2 Shaping fonts

The digitized characters produced by designers must be shaped in order to ensure optimal rendition on various visualization and printing devices. The shaping process may, for instance, mean ensuring that stems all have the same width and that serifs do not vary from one character to another. The slight variations due to the digitizing process must be removed. In order to ensure maximal character rendition at both low and medium resolution, hints must be added to character outline descriptions. Last, the fonts are grouped into families and styles and copied onto the production disks before being marketed.

[7] Several forthcoming publications of the Didot Project will be devoted to this particularly complex issue. Let us note here Chuck Bigelow's statement that 7 years' study are required to become a good typographer [17].

2.3 Handling fonts

An end-user working on a PC or a workstation must be able to handle his fonts according to their family, their style, their height of type and their computing functionalities. A desktop publishing specialist needs bitmap fonts as well as outline fonts in various formats (for instance, *Ikarus, Adobe Type I* and *True Type* formats). When necessary, the fonts will be converted from one format to another or used to create bitmap fonts with a given height of type. Some companies (publishers, software houses, laboratories, etc.) as well as font distributors handle and service a large number of fonts and their associated software. They therefore require databases, great storage capacity, several programs for converting between font formats, adequate data access procedures and automatic processing capabilities, all of which call for modern software engineering techniques.

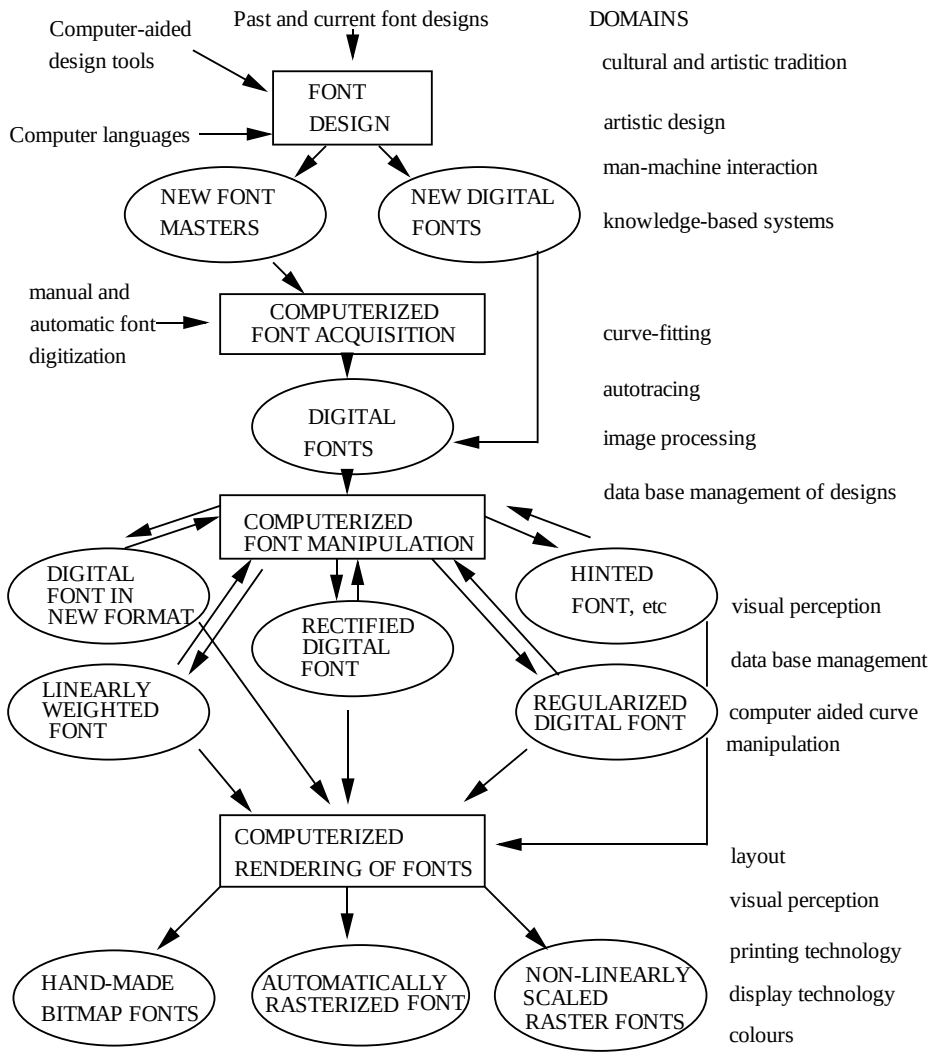

Figure 2. Main fields of activity in digital typography

2.4 Calling fonts

For the end-user, every font is associated with a given name and height of type. The system also links it to metric properties (for instance the typesetting tables used to prepare justification) and often to a font representation which can be used on the screen.

2.5 Font rendition on various devices

Identical computer fonts must be rendered optimally on very different supports: visual display units, phototypeset films, paper printed by a page printer, king-size display units in sports stadiums and airports, inclusion in video films, etc. Each device alters the appearance of the original characters in a different way. Therefore, the rendering software must take all these alterations into account and try to compensate in order to obtain the best rendition possible.

2.6 Research and development of new applications

Computer scientists and programmers who develop software for the acquisition (scanning, autotracing), the shaping (uniformization, addition of hints) and the rendering (scan-conversion, compensation of device characteristics) of characters need tools which enable them to individually access different character parts according to their requirements. There may also be a need for analyzing font databases, for instance to carry out statistical studies on character shapes and metrics. This is why tools offering access to both metric [19] and topological [20,21] font properties must be used.

The various procedures related to font creation, acquisition, shaping and rendition are described in Figure 2. It also lists the fields which must be mastered when developing programs capable of performing the aforementioned processes.

3 A STUDY OF THE NEED FOR TEACHING DIGITAL TYPOGRAPHY

Let us first try to determine at whom a curriculum or a specialization course in digital typography is aimed. It mainly concerns all those involved in creating, shaping, handling and rendering fonts and who have already completed a prior scientific and computer science training. The various fields of activity can be defined as follows:

1. *Firms* that assemble computers, laser printers, high-speed printers, teleprinters, phototypesetters and telecommunication terminals as well as those who manufacture specialized display and printing devices are concerned with digital typography. They produce and market hardware and software for creating and manipulating texts.
2. Companies offering *specific services and products* for typography. Many of them market an ever-increasing number of fonts in various industrial formats [22] (examples: Linotronic, Agfa-Compugraphic, URW, Adobe, Autologic, Mecanorma, etc.). Other firms such as URW, Letraset, Altsys or Typographics offer programs for creating and modifying fonts. Some companies have artistic design programs that include typographic features (Adobe *Illustrator*, Aldus *Freehand*, etc.).

3. Designers involved in producing *interactive programs* in a variety of fields (mechanical CAD, architecture, etc.) wish to include in their product facilities for selecting fonts, displaying texts and printing documents. They therefore purchase the typographic facilities distributed by computer companies (*TrueType* on Macintosh and PC Windows, Adobe *Type Manager* on Macintosh). Moreover, they include facilities for producing documents in PostScript format which can therefore be printed or displayed on various printing devices (laser printer, photocomposer, *PageView* on Sun or *DisplayPostScript* on IBM Risc 6000).
4. The most common type of firm, and moreover those who feel the greatest need for basic training in digital typography, are those who provide document *phototypesetting and printing services.* These companies receive documents which have been created using a number of different programs (*PageMaker, Word, QuarkXPress, WritePerfect, TEX,* etc.). Such documents include both text and illustrations. They must be revised before printing so that the fonts they include correspond to those that are available. Documents sometimes have to be typed in once again before they are sent to be flashed (PostScript phototypesetter). Handling fonts and approximating fonts with similar ones requires in-depth knowledge of letterpress typography as well as computer know-how. The page layout program must be sent the right spacing tables and documents written in PostScript language must be adjusted, altered and tested. In order to master such a job, normally trained computer technicians need to follow postgraduate courses in digital typography and related fields.

Listing the various fields of economic activities shows that there are many people who require training in digital typography. The structure and the content of such courses should be varied enough to include both training programs aimed at those who wish to train while holding a job and courses tailored to meet the needs of the various aforementioned categories.

A one-year full-time specialization course in digital typography aimed at a small number of engineers and highly qualified teachers could also be set up. This would help train teachers for technical colleges (packaging and paper engineering schools, graphic arts technical schools, etc.) as well as engineers employed by *companies* producing computers, displays or printing devices.

4 TEACHING FIELDS

Let us try and give a brief description of the subjects which could—or should—be included in a curriculum for digital typography.

Bibliography: Aside from the main subjects included in the curriculum, we will list a series of bibliographical notes about the subject aimed at those who wish for more information. Most of them are made up of research papers. Those marked with an asterisk (*) are mainly devoted to teaching methods or basic training.
There is no handbook on digital typography. However, interesting information can be found in [13]*, [23]*, [24]*, [25], [26], [27] and [7]*.

Tools: In a similar manner, we will point out the few pedagogic tools that are available or currently being written.

4.1 Visual perception

Aims: Character outlines are always defined taking into account how they will be perceived by the human eye. Digital typographers must be familiar with vision and legibility problems.

Subjects: Physiology and psychology of vision, statistical models, knowledge models, models based on signal theory.

Bibliography: [28] [29] [30] [31] [32]*.

4.2 Character typology and topology

Aims: Notwithstanding some "subjective" aspects (such as the notion of style), characters (taken individually) and fonts (taken globally) display various metric properties (height, angle of italics, etc.) or topological properties (for instance, presence and shape of serifs). These properties should be explicitly described in digitized fonts since know-how about such properties may enhance character rendition.

Subjects: Anatomy of letters, terminology, font metrics, non-linear scaling, units, font description models, standards.

Bibliography: [33] [19]* [34] [35] [36] [20] [37] [38] [39] [23]* [18] [15] [25].

4.3 Mathematical description of forms

Aims: Characters may be described in different ways [40]: either by pen displacement [41], by structural elements [42] or, as is usually the case, by outline description. Pen displacements and outline description can be described mathematically through analytic curves. Prior knowledge of vector analysis is required. Character description implies sound knowledge in computer-aided geometric design.

Subjects: Polynomial curves, interpolation, splines (natural, Bézier and B-splines).

Bibliography: [43]* [44]* [45]* [22]*.

4.4 Printing and display technology

Aims: Almost all printing or display devices are based on rasterized images and include one or several grey levels whose properties must be mastered.

Subjects: Rasterization techniques, printing techniques (laser, phototypesetting), halftoning techniques, HDTV, greyscales, moiré pattern formation.

Bibliography: [46] [47] [48]* [49]*.

4.5 Page description languages

Aims: The objects and characters to be rendered on a raster device (see Section 4.4) are often described using page description languages.

Subjects: Page description languages, PostScript, SPDL, standards.
Bibliography: [50] as well as the many other available books about PostScript.
Tools: PostScript Previewers, for example [51].

4.6 Rendering characters on display and printing devices

Aims: Characters are usually rendered on raster devices. This means taking into account the characteristics of the output device (screen, printer; see Section 4.4) as well as the main characteristics of the human visual system (see Section 4.1). Knowledge of 2-D computer graphics and image processing is also required (see Section 4.3).
Subjects: Adapting outlines to the grid, scan conversion, filling algorithms, filtering, resampling, spatial-frequency analysis in the Fourier domain.
Bibliography: [52] [53]* [54]* [47].

4.7 Internal font management

Aims: Fonts are complex structured data that must be managed using software engineering techniques (see Section 2.3).
Subjects: Font formats (see Section 4.2), format changes, databases, networks and transmission techniques, hypertext font systems, font servers.
Bibliography: [33] [34] [55] [23] [56] [57]. See also Section 4.2.

4.8 Entering and producing digitized fonts

Aims: The general public has only very recently gained access to hundreds of digitized font families. This is due to the large amount of quite tedious work required to enter a great number of fonts and shape them in order to make them suitable for use on screens, printers and phototypesetters. Fonts can be entered either by digitizing their outline manually using a graphic pad [23] or by digitizing them on a scanner and extracting their outline automatically. It has not yet been possible to computerize the different processing steps that ensure that input or digitizing errors are corrected, that characters are shaped by regularizing their stems and serifs and that appropriate hints are added. Each new font includes one or two characters that do not comply with the possibilities of automatic software processing and therefore require manual treatment. Methods for regularizing outlines and adding hints require using pattern-recognition techniques [58] and character analysis based on topological and geometric criteria.
Subjects: Digitizing, outline extraction, approximating outline parts using polynomials, regularizing outlines, shape recognition, geometric and topological knowledge.
Bibliography: [59] [60] [58] [22] [20] [18].
Tools: *Interactive Rastware*: program for interactive hint creation, manipulation and visualization developed at EPFL.[8]

4.9 Other items

Aims: Digital typography is not a rigid field but one that must evolve. Teaching must therefore also include a look at neighboring application fields or subjects that are still being researched. At the moment, we can include the following items:

Subjects and bibliography:

- Creating characters [40]
- Font quality [61] [62]
- Character recognition [63] [59] [64]
- Incremental font development, dynamic fonts [21] [65] [66]
- Copyright [17] [67]

5 CONCLUSION

A course involving all described subjects has not been tested as a whole. However, various items have been used either for specific courses (among others at the Peripheral Systems Lab, EPFL Lausanne), for conferences (for instance, during various courses at Ifsic in Rennes), or during full-time training sessions (for instance, at the Didot School in Lausanne).

The experience of the Didot School in Lausanne (see Section 1) has shown that there is a real need for training in digital typography. Such training should be aimed at engineers who develop new products related to typography and printing as well as at people who offer services for creating, handling and printing documents using software and hardware available on the market.

REFERENCES

1. Robert L. Ashenhurt, ed., 'Curriculum recommendations for graduate professional programs in information systems', *Communications of the ACM*, **15**(6), 363–398, (May 1972).
2. J. Daniel Couger, ed., 'Curriculum recommendations for undergraduate programs in information systems', *Communications of the ACM*, **16**(12), 727–749, (December 1973).
3. Mary Shaw and James E. Tomayko, 'Models for undergraduate project courses in software engineering', in *Software Engineering Education*, 33–71, (1991). Proceedings of the SEI conference, Pittsburgh, Pennsylvania, USA, October 7–8, 1991.
4. Jacques André and Roger D. Hersch, 'DIDOT European project for training in digital typography', *TEXline*, **10**, 21–22, (1991).
5. Mary Dyson, 'Update on Didot', *TEXline*, (14), 20, (1992).
6. Mary Dyson, 'Update on Didot', *Newsletter of the British Computer Society Electronic Publishing Specialist Group*, **7**(1), 5–6, (1991).
7. *Visual and Technical Aspects of Type*, ed., R. D. Hersch, Cambridge University Press, 1992. To be published.
8. Piero de Macchi, 'La prima scuola estiva europea di tipografia digitale', *graphicus*, (882), 49, (January 1992).
9. Richard Southall and Jacques André, 'Experiments in teaching METAFONT', in *TEX for Scientific Documentation*, ed., D. Lucarella, 141–153, Addison-Wesley, 1985.

[8] This program and associated documentation can be requested for a simple handling fee from the Peripheral Systems Lab, EPFL.

10. Thomas Detrie and Eugene Valentine, 'The workshop as format for letterform design education/ das seminar als modell für die ausbildung in schriftgestaltung', *Swiss Typographic Journal, Typografische Monatsblätter, Revue Suisse de l'Imprimerie*, **TM6**, 2–24, (1989).
11. *Didot bulletin*, n° 1, May 1991.
12. *Didot bulletin*, n° 2, July 1992.
13. Richard Rubinstein, *Digital typography—an introduction to type and composition for computer system design*, Addison-Wesley, Reading, MA, 1988.
14. Fernand Baudin, *La typographie au tableau noir*, éditions Retz, Paris, 1984.
15. Walter Tracy, *Letters of credit—a view of typed design*, Gordon Fraser, London, 1986.
16. Gérard Blanchard, *L'eredita Gutenberg—per una semiologia della tipografia*, Gianfranco Altieri editore, Italy, 1989.
17. Chuck Bigelow, 'Notes on typeface protection', *Tugboat*, **7**(3), 146–151, (1986). Also: 'Du piratage des fontes', *TSI Technique et science informatique*, vol. 6, 3, 1987, 255–259.
18. Donald E. Knuth, *The* METAFONT *book*, Addison-Wesley, Reading, MA, 1986.
19. Jacques André and Justin Bur, 'Métrique des fontes postscript', *Cahiers GUTenberg*, (8), 29–56, (March 1991). To appear as 'Font metrics' in [7].
20. Roger D. Hersch and Claude Bétrisey, 'Model-based matching and hinting of fonts', *Proceedings SIGGRAPH'91, ACM Computer Graphics*, **24**, (1991).
21. Debra Adams, 'abcdefg, a better constraint driven environment for font generation', In André and Hersch [26], 54–70.
22. Jakob Gonczarowski, 'Curve techniques for auto-tracing', In Hersch [7]. To appear.
23. Peter Karow, *Digital Formats for Typefaces*, URW Verlag, Hamburg, 1987.
24. Peter Karow, 'Digital punch cutting', *Electronic Publishing: Origination, Dissemination, and Design*, **4**(3), 151–170, (September 1991). To appear as a chapter of P. Karow, *Font Rasterization.*
25. Alison Black, *Typefaces for desktop publishing—a user guide*, Architecture Design and Technology Press, London, 1990.
26. *Raster Imaging and Digital Typography*, eds., Jacques André and Roger D. Hersch, Cambridge University Press, 1989.
27. *Raster Imaging and Digital Typography II*, eds., Robert A. Morris and Jacques André, Cambridge University Press, 1991.
28. *Eye movements and their role in visual and cognitive processes*, ed., E. Kowler, Elsevier Science Publishers BV, 1990.
29. Guozhen Duan and Robert A. Morris, 'The importance of phase in the spectra of digital type', *Electronic Publishing: Origination, Dissemination, and Design*, **2**(1), 47–59, (1989).
30. Gordon E. Legge, Gary S. Rubin, and Andrew Luebker, 'Psychophysics of reading', *Vision Research*, **25**(2), 239–252, (1985).
31. Robert A. Morris, 'Image processing aspects of type', in *Document manipulation and typography*, ed., J. C. van Vliet, 140–155, Cambridge University Press, 1989.
32. *Recherches actuelles sur la lisibilité*, ed., François Richaudeau, éditions Retz, Paris, 1984.
33. Adobe Systems Incorporated, *Adobe Type 1 Font Format*, Addison-Wesley, Reading, MA (USA), 1990.
34. Daniel Dardailler, 'Normes et fontes', *Cahiers GUTenberg*, **4**, 2–8, (1989).
35. Jakob Gonczarowski, 'Industry standard outline font formats', In Hersch [7]. To appear.
36. Bernd Holthusen and Albert Jan Pool, *Scangraphic Digital Type Collection*, Mannesmann Scangraphic, Wedel/Hamburg, 1990.
37. *ISO/IEC DIS 9541, Information processing—font and character information interchange*, 1991.
38. Mark Jamra, 'Some elements of proportion and optical image support in a typeface', In Hersch [7]. To appear.
39. Marcel Jacno, *Anatomie de la lettre*, École Estienne, Paris, 1974.
40. Richard Southall, 'Character description techniques in type manufacture', In Morris and André [27], 16–27.
41. Donald E. Knuth, 'The concept of a meta-font', *Visible Language*, **XVI**(1), 3–27, (1982).
42. Pierre Coueignoux, *Generation of roman printed fonts*, Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, 1975.
43. R. H. Bartels, J. C. Beatty, and B. A. Barsky, *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling*, Morgan Kaufmann, Los Altos, CA, 1987.

44. Carl de Boor, *A Practical Guide to Splines*, Springer Verlag, New York, 1978.
45. Gerald Farin, *Curves and Surfaces for Computer Aided Geometric Design*, Academic Press, San Diego, CA, 1982.
46. Isaac Amidror, 'The Moiré phenomenon in color separation', In Morris and André [27], 98–120.
47. Avi C. Naiman, *The Use of Grayscale for Improved Character Presentation*, Ph.D. dissertation, University of Toronto, 1991.
48. William M. Newman, *Principles of Interactive Graphics*, McGraw-Hill, 1979.
49. Robert Ulichney, *Digital Halftoning*, MIT Press, 1987.
50. Adobe Systems Incorporated, *PostScript Language Tutorial and Cookbook*, Addison-Wesley, Reading, MA (USA), 1987.
51. Catherine André and Claude Bétrisey, 'An enhanced PostScript previewer for experimenting and teaching new approaches in digital typography', in *Teaching Electronic Publishing*, ed., Jacques André, 83–95, (1992). Proceedings of TEP'92, Lausanne, 1992. Appeared as *Bigre* 79.
52. R. D. Hersch, 'Introduction to font rasterization', In André and Hersch [26], 1–13.
53. David Rogers, *Procedural Elements for Computer Graphics*, McGraw-Hill, 1985.
54. Gideon Avrahami and Vaughan Pratt, 'Sub-pixel edge detection', In Morris and André [27], 54–64.
55. Roger Debry, Alan W. Griffee, and James P. Hofmeister, 'Multiple font technologies in a distributed environment', In André and Hersch [26], 223–231.
56. Bernard Marti, Hervé Layec, and Christiane Schwartz, *Télématique, techniques, normes, services*, Dunod, Paris, 1990.
57. Edwin J. Smura, Barbara Beeton, Karla Savage, and Alan Griffee, *Font Information Interchange Standard ISO/IEC9541, Computer Communications*, April 1989.
58. Peter Karow, 'Automatic hinting for intelligent font scaling', In André and Hersch [26], 232–241.
59. Theo Pavlidis, *Structural Pattern Recognition*, Springer Verlag, 1977.
60. Michael Plass and Maureen Stone, 'Curve fitting with piecewise parametric cubics', *ACM Computer Graphics*, **17**(3), 229–239, (July 1983).
61. Ittai J. Tamari, 'Decipherability, legibility and readability of modern Hebrew typefaces', In Morris and André [27], 128–136.
62. Debra Adams and Richard Southall, 'Problems of font quality assessment', In André and Hersch [26], 213–222.
63. *Proceedings of the 1st International Conference on Document Analysis and Recognition*, ed., Guy Lorette, AFCET publ., Paris-Rennes, 1991.
64. Abdel Belaïd, ed., 'Traitement de l'écriture et des documents', *Bigre*, (80), (July 1992). Special issue.
65. Debra Adams and Jacques André, 'New trends in digital typography', In André and Hersch [26], 14–21.
66. Erik van Blokland and Just van Rossum, 'Different approaches to lively outlines', In Morris and André [27], 28–33.
67. André Bertrand, 'La typographie et la loi', *Cahiers Gutenberg*, (8), 10–20, (1991).