ELECTRONIC PUBLISHING, VOL . 4(3), 151–170 (SEPTEMBER 1991)

Digital punch cutting

PETER KAROW

URW
Harksheider Straße 102
2000 Hamburg 65
GERMANY

SUMMARY

Digital punch cutting is today's font technology. There are three different methods available for getting alphabets into digital form : hand-digitizing, auto-tracing and direct design on a workstation screen. The advent of intelligent font scaling requires us to ensure the 'optical' quality of a font and also the 'numerical' quality of its data; this, in turn, means that new procedures have to be added to the font production process. Furthermore, a given typeface has to be rendered on a wide variety of output devices ranging from computer displays, printers (dot-matrix, laser, inkjet or thermal-transfer) and typesetters (CRT or laser) to the more exotic devices such as plotters, vinyl-cutters and routers. To deal with this it is necessary to set up a database of font data, in a machine-independent format such as IKARUS. This enables us to cope with the long life cycles of typefaces and also to serve present and future applications by converting the IKARUS data into various machine-specific formats.

KEY WORDS Digital typefaces Hand-digitizing IKARUS format Auto-tracing Font technology Intelligent font scaling

1 INTRODUCTION

It is becoming ever more apparent that we at URW are treading a new path in the handling of typefaces with the IKARUS system. In particular, the method of *hand-digitizing* plays an important role in this field; but this is a technique which seems difficult for typographic 'insiders' to justify and for 'outsiders' to understand (Figure 1).

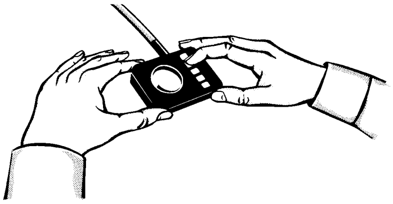


Figure 1. Sensor control is done best with two hands

There are two fundamental, psychological reasons that hinder our clear comprehension of this method. On the one hand we have scanning as a competitive, automatic method of digitizing but on the other hand there is a certain tendency to idealize our expectations of this technique and to ignore the realities — in actual fact the automation

0894–3982/91/030151–20$15.00

Received 8 February 1990
Revised 15 August 1990

can never be as perfect as we imagine. Therefore, attracted by such automation and neglecting the existence of hand-made, imperfect artwork, we underestimate the value of hand-digitizing. We deal with both of these aspects in the next two sections and address the question of 'designing on the screen' in the third section.

Another advantage of hand-digitizing, in conjunction with IKARUS technology, is the attainable quality, and we devote the fourth section to this. In a final section we make clear the importance of dividing the work between the creation of a typeface database and the manufacture of type in various machine-specific formats.

2 HAND-DIGITIZING

As well as IKARUS, there are other tools on the market for digitizing and manipulating fonts. These include Fontographer[1] and FontStudio[2] for outlines and Adobe Type Manager (ATM)[3] and Fontware[4] for bitmaps using intelligent scaling. Further background information about digitizing can be obtained from the papers by M. Stone [1] and L. Ruggles [2].

The remainder of this paper is based primarily on experience with IKARUS in those areas where software plays a role. IKARUS is the only existing software which allows hand-digitizing. This fact leads some people — perhaps those not familiar with it — to contend that hand-digitizing is tedious and inexact compared to scanning. In general, our experience is entirely to the contrary. At URW we use both methods and decide on a case-by-case basis which technique to employ when digitizing a font.

We have the LINUS system at URW which is one of the best products on the market for generating outlines based on scanned patterns; our assertions are based on this method of outline generation. Products such as Streamline from Adobe Systems or Free Hand from Altsys/Aldus[5] would point us similarly in the direction of hand-digitizing.

In order to avoid misunderstandings, what we propose to address is the whole area of 'font technology': the creation of a typeface by exploitation of the most up-to-date technology and by use of the most advanced know-how and findings. We shall also address production issues, i.e. effectiveness and quality, from an economical point of view.

Digitizations are composed of points characterized by their coordinates and their type. There are three different types: *corner*, *tangent*, and *curve* points. A fourth type — the *starting* point — serves to indicate the start of closed contours (outlines). For example, there are two contours for the letter 'O' and just one for the letter 'W'.

One can picture an outline as a sequence of straight and curved lines. Two joined lines or two joined curves meet each other at an angle. Their intersection is where we find the first kind of digitization point, the *corner* point. (Note that if the lines or curves happen to merge 'flush' into one another, they can then be combined into a single line or curve, respectively.)

---

[1] Fontographer is a product of Altsys Corp., 269 W. Renner Road, Richardson, TX 75080, USA.
[2] FontStudio is a produce of Letraset, Esselte American Operations, 40 Eisenhower Drive, Paramus NJ 07653, USA.
[3] ATM is a product of Adobe Systems Inc., 1585 Charleston Road, P.O. Box 7900, Mountain View, CA 94039-7900, USA.
[4] Fontware is a produce of Bitstream, Bitstream Inc., Athenaeum House, 215 First Street, Cambridge, CA 02142, USA.
[5] Aldus Corporation, 411 1st Ave. South, Suite 200, Seattle, Washington 98104, USA.
Altsys Corp., 269 W. Renner Road, Richardson, TX 75080, USA.

The second kind of digitization point is found on curves and is therefore called a *curve* point. These serve as supports and are positioned wherever the direction of the curve has changed by approximately 30 degrees. Additionally, all local X and Y extreme values of outlines are digitized with curve points. For example, on a circle these points are the outermost left, right, upper, and lower points.

We call the third kind of digitization point the *tangent* point. This is used wherever a straight line turns smoothly (tangentially) into a curve, or vice versa. Here, the line and curve lying to the left and right of the tangent point have the same direction, as shown in Figure 2 (see also Reference [3]).

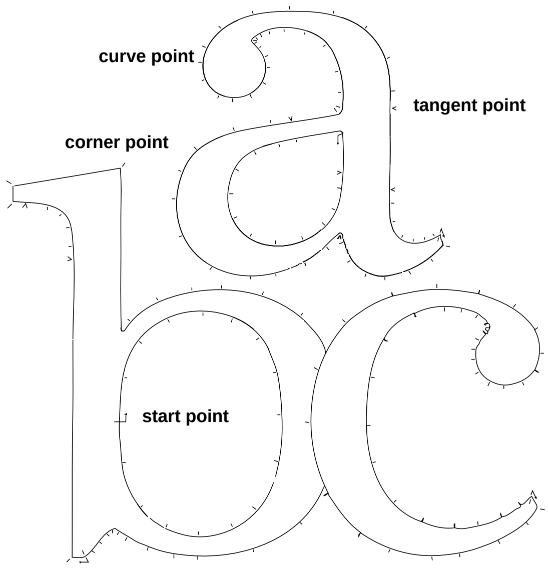


Figure 2. Illustrative example of digitizing; shown are characters marked with the IKARUS method

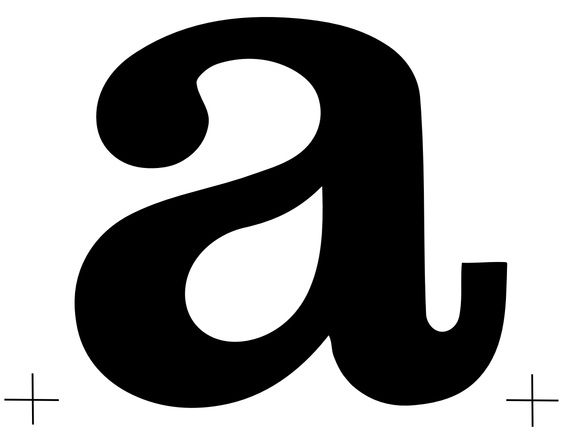


Figure 3. Artwork for an ‘a’*, a good example which could be digitized without further checking*

It has been very instructive for all of us to have the alternative of scanning at our disposal, in addition to hand-digitizing. In Table 1 we give some statistics which sum up our eighteen years of experience, based on the following preconditions:

- A typeface has 100 characters, each composed of approximately 50 IKARUS points (IK-points). This is around 5000 digitized points (digs).
- During digitization, good artwork is used, having a quality similar to that of the letter ‘a’ in Figure 3.

Hand-digitizing yields an ‘a’ of a quality similar to that shown in Figure 4.
Scanning and conversion into an IK-contour produces the quality shown in Figure 5.

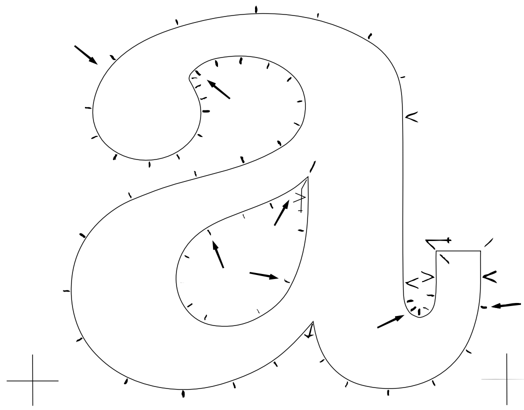


Figure 4. Approximately 10% of the IK-points are inaccurate after hand-digitizing

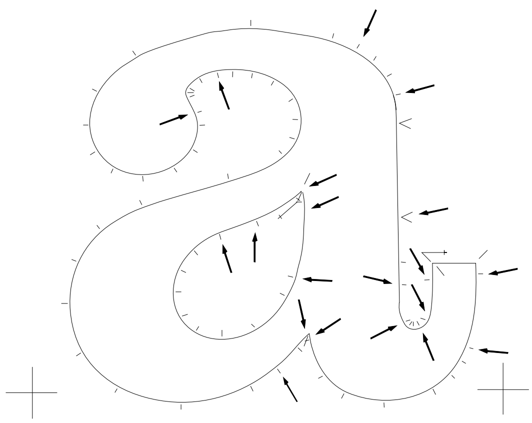
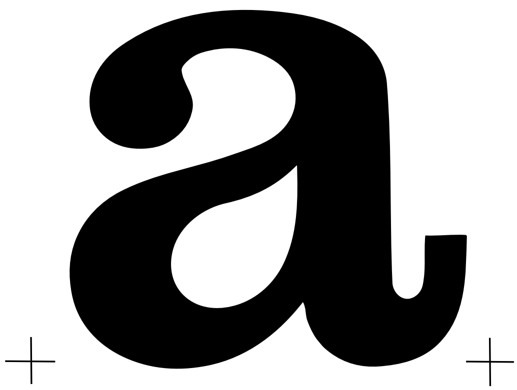
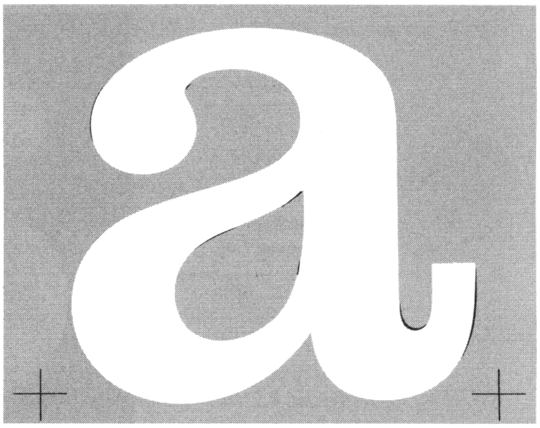


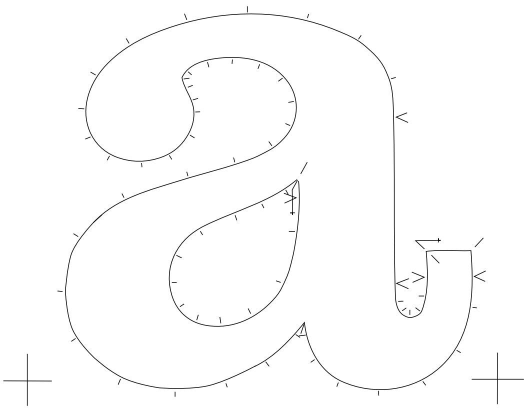
Figure 5. Approximately 30% of these scan-generated IK-points are inaccurate after auto-tracing



(a) Final shape



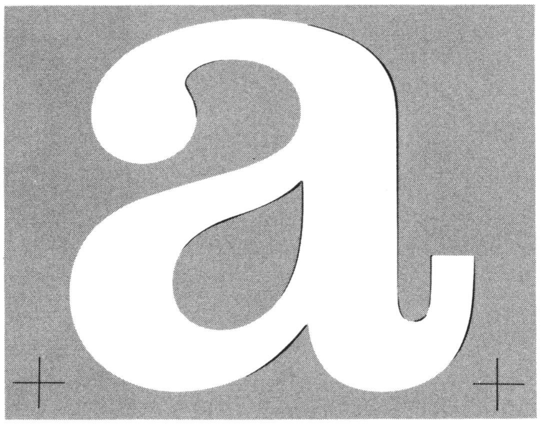
(b) Changes are marked in black



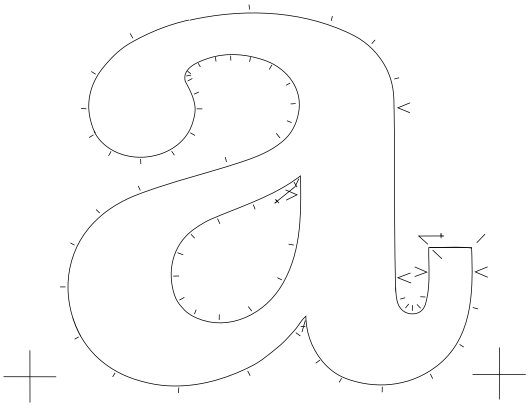
(c) Final IK-points

Figure 6. Comparison of the hand-digitized result with the original

(a) Final shape



(b) Changes are marked in black



(c) Final IK-points

Figure 7. Comparison of the auto-traced result with the original

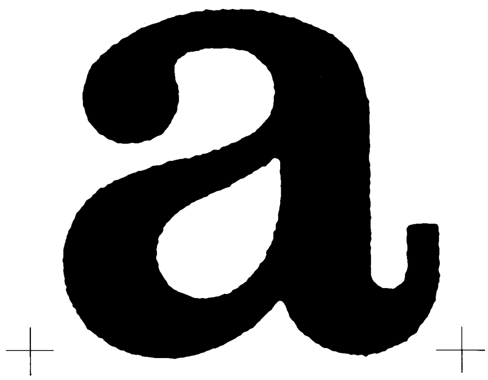


Figure 8. Example of an ‘a’ *at 24 pt greatly enlarged onto glossy paper*

Table 1. Comparative work expenditures for hand-digitizing and for auto-tracing

Work	Hand-digitizing	Auto-tracing
Data input		
Times for input	4 sec/dig ⇒ 5.5 hrs/typeface	0.1 sec/dig ⇒ 0.1 hrs/typeface
Operating	90 sec/char ⇒ 3 hrs/typeface	90 sec/char ⇒ 3 hrs/typeface
Mistaken digitizations	10%	30%
First correction		
Times for interactive improvements	30 sec/dig ⇒ 4.2 hrs/typeface	30 sec/dig ⇒ 12.5 hrs/typeface
Technical refinements (Second correction)		
Channel processing	2 hrs/typeface	4 hrs/typeface
Serif cut/paste	included in hand-digitizing	5.5 hrs/typeface
Symmetrizing	2 hrs/typeface	5 hrs/typeface
Setting extrema, final design improvements	2 hrs/typeface	4 hrs/typeface
Operating	2.7 hrs/typeface	2.7 hrs/typeface
Total	21.4 hrs	36.8 hrs

The overview of work expenditures summarized in Table 1 refers to the production of an ensemble of 100 Latin letters.

After interactive corrections on the screen, the following results are obtained from hand-digitizing (Figure 6) and from automatic digitizing (Figure 7).

The amount of work required for scanned typefaces increases drastically when one digitizes artwork of the kind shown in Figure 8 (this is an example which could be digitized only after researching and preparing a suitable digitizing scheme).

When pencil drafts are used as input originals (and this is not uncommon) scanning can be completely disregarded. By using pencil drawings, designers can save themselves approximately 30% to 50% of the effort needed with other methods; this kind of original is the preferred choice with the existing IKARUS system.

3 ORIGINALS

Typeface manufacturing embraces alphabetic characters including Kanji and hieroglyphs, as well as signets, logos, symbols and pictographs (line art and photographs are not included). In other words, the elements of setting — text composition — are dealt with under the general concept of typeface manufacturing. The elements of page layout — its typography — such as finished texts, borders, graphics, and pictures, however, are not included (Figure 9).

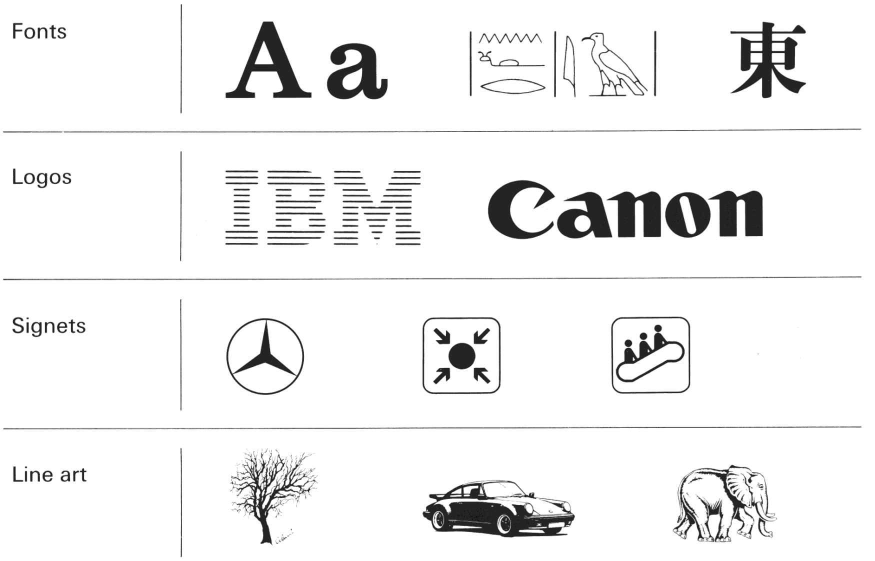


Figure 9. Originals for digitizing are not restricted simply to typefaces!

As a general rule, typefaces are not only drafted but are also digitized and constructed as well. The more difficult the degree of construction, the more likely we are to digitize by hand, with the assistance of the usual technological drafting equipment such as a curve template or a ruler. Only a few typefaces such as ‘Block’ or brush lettering (e.g. the Chinese Kanji style ‘**Li Shu**’) have contours which do not need these aids. The decision of whether to digitize by hand or with the scanner (which generates an automatic outline) depends solely on the desired quality for the outline and not on the nature or the appearance of the typeface.

For purposes of illustration, we have reproduced details from randomly chosen artwork that we have been working on recently (Figures 10, 11 and 12).

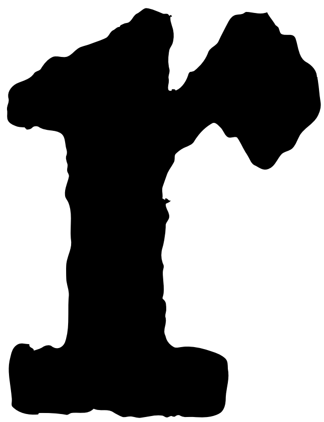


Figure 10. Golden Type, enlarged view of a 12 pt ‘r’. *Approximately 50 pencilled characters must be added to the photographed originals*

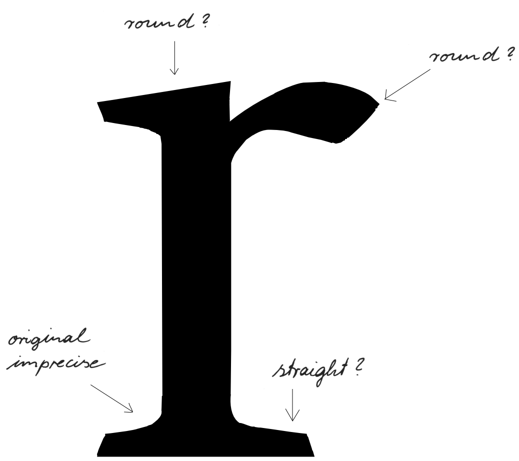


Figure 11. ITC Giovanni, output on laser printer (with 30 characters added)

As a rule, one strives for a design of high quality when creating typefaces but achieves this only partially in the draft. Because of the limitations of our senses, all humans have minor problems recognizing what the designer of a typeface really wanted. Herein lies the preference for hand-digitizing — it enables us to discern the kind of technical quality that the designer was striving for. By way of example, many of the preceding figures have small arrows indicating places of potential difficulty that prove to be no problem at all once we have decided on the correct position of points along the contour during digitization.

During hand-digitizing a draft is transferred, in its technical form, to the computer. In this sense, the designer at a computer is comparable to the punch cutter of yesterday.

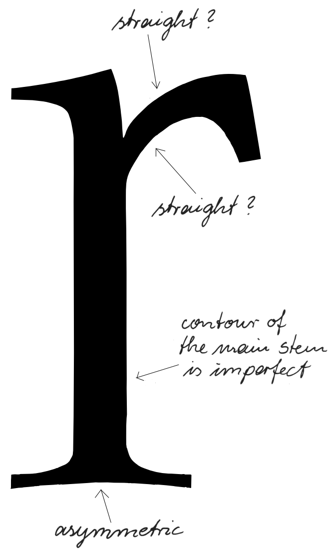


Figure 12. Weidemann Antiqua. Artwork in ink on heavy paper

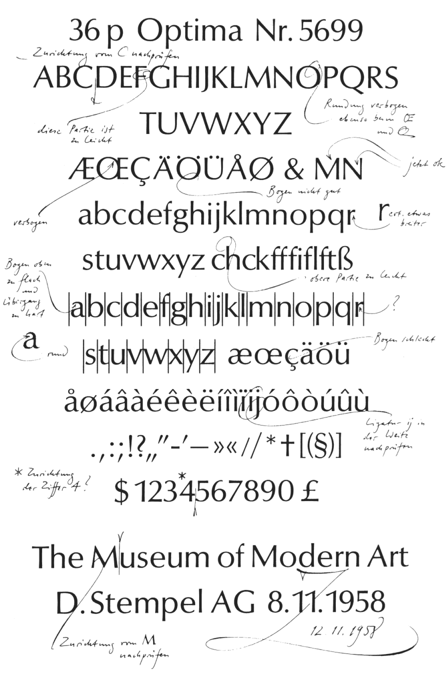


Figure 13. Hermann Zapf's notations on his Optima typeface

The draft receives its technical, and structural perfection through digitizing. Therefore, it is no wonder that even specialized typeface designers require approximately 100 – 200 hours for a good digitization of about one hundred characters. These computer specialists have to know not only what the designer wanted but also what today's machine typesetting requires.

It is instructive to read through the annotations that Hermann Zapf made on his Optima typeface (Figure 13) which experts find to be fully sufficient for conveying the desired effects. This kind of communication between designers and typeface producers continues to take place today, just as it did thirty years ago [4]. Technical expertise of this sort used to be commonplace at the well-known type foundries and is still achievable today in some of the better typeface divisions set up by the manufacturers of typesetting machines. Firms such as URW, which specialize in typeface manufacturing, can also produce work of this quality. The general public is often unaware of the skills involved in transforming a draft into a typeface. Digital artwork has to be produced with typeface know-how — not just with a curve template and a ruler but also with a steady hand and a digitizing tablet. At a later stage this artwork is transferred from the screen onto paper and onto film (either on a plotter or a typesetter). All of this requires an interactive dialogue with the computer based on the computer's own, precise, hard-copy output.

4 DESIGNING ON THE SCREEN?

In the future shall we be moving to a stage where new typefaces are drafted solely with the help of workstation screens and computers? Donald Knuth's Metafont program, for

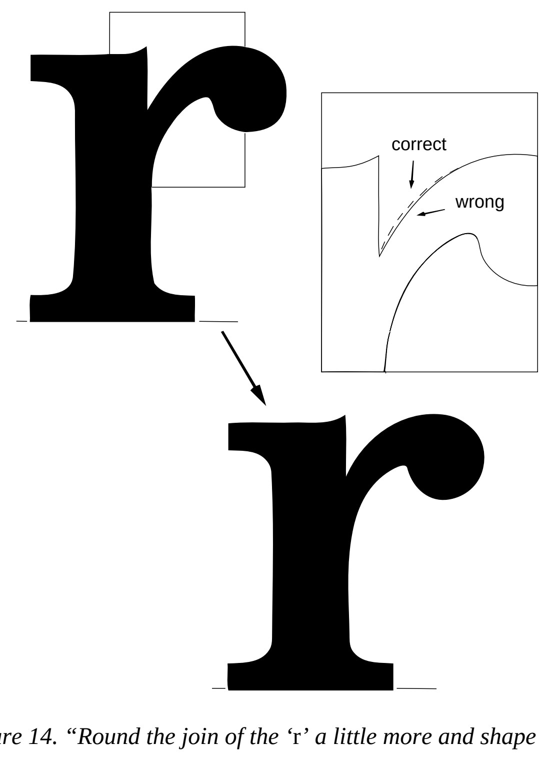


Figure 14. "Round the join of the 'r' *a little more and shape it correctly"*

typeface design [5, 6], already uses this method, but we have some reservations about it. Our experience with computers has taught us that using this method alone does not suffice when transcribing *existing* typefaces into the digital world. The tools we need, as well as a digitizing tablet, are a precise (flatbed) drafting machine and a typesetter. Only then can we compare the resulting letters with the originals.

It is interesting to see what happens when a lower-resolution proofing device is used, for example, by comparing an outline drawing from a laser printer with the original. The output height of 10 cm letters fluctuates up to half a millimetre, depending upon the day-to-day performance of the laser printer, so how can one ensure uniformity of size? Certainly one can regulate the proportions through numerical proofing and correction, with help from the screen. However, this does not allow comparison with the original, except when the original has been automatically scanned. On the other hand, this scanning is itself executed to an accuracy of only 300 lines per inch (i.e. with an error of ±0.1 mm). For the reproduction of typefaces, this is not acceptable (see final section). If one digitizes an existing typeface using DTP-quality scanners and laser printers then the kindest description that can be given to the result is that it represents a 're-design' of the original.

But what is the situation for a genuinely new draft? A designer could certainly try his luck with the screen alone but our experience at URW is that our designers would happily prefer to reach for their pencils and do a hand-drafted outline, which will be input to the computer via a digitizing tablet in the end. The digitizing process takes only two to five minutes per character but the draft itself can take 20 – 60 minutes if it is created with

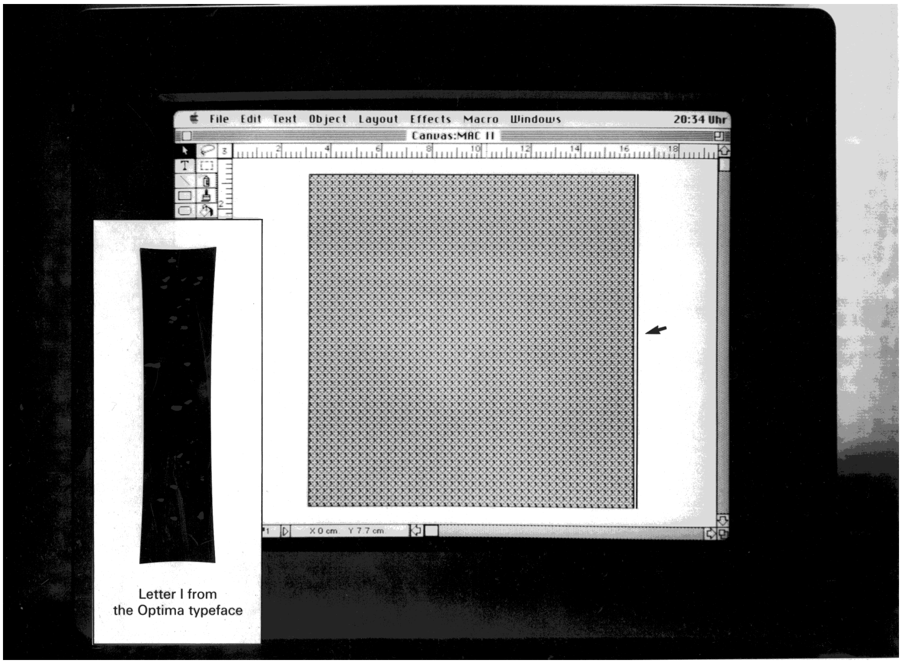


Figure 15. All screens display a 'pillow distortion'. Hermann Zapf could not have designed the Optima typeface using a screen

pencil and paper; if the draft is originated on the computer it takes twice as long. A further example of relative timescales is given by the operation described in the caption to Figure 14 (which concerns a small modification to the letter ‘r’). Using graphical methods the operation takes about 10 seconds; interactively, on the screen, it needs more than 40 seconds.

Another important factor is that screens cannot give a distortion-free picture. For example, Figure 15 shows ‘pillow’ distortion in a square which has been defined correctly from a numerical point of view; if you are imaginative enough, you can convince yourself that the screen really does display a square! In consequence, we employ screens for interactive revision only. We establish the values for corrections using precise drawings from the drafting machine, based on comparisons with the original. When we *know* what we want, we can execute it on the screen. As stressed at the beginning, we are addressing the problem of technical conversion of original drawings, in the same sort of way that a punch cutter needed to take careful measurements to establish exact lines and smooth curves. We particularly need to know accurate sizes for any necessary corrections and these can only be achieved by comparison and measurement using reliable tools.

5 QUALITY

What degree of precision is ultimately required? For us, there is no longer any doubt about the answer to this question. Over the course of the past eighteen years, in contact with all typeface divisions of the world, we have determined that a practised eye can detect discrepancies in curves of a magnitude in the region of ±0.03 mm. We assume that this accuracy is entirely adequate for straight lines and indeed it can be achieved without difficulty using today’s technology. However, although the representation of straight lines does not need to be discussed any further, the quality of curves certainly does (Figures 16 – 19).

It turns out, in hand-digitizing, that a body size of 15 cm is the easiest to work with and that an accuracy of at least ±0.03 mm must be reached. Therefore, we decided to aim for a technical resolution of ±0.01 mm, which means 15 000 × 15 000 units per em

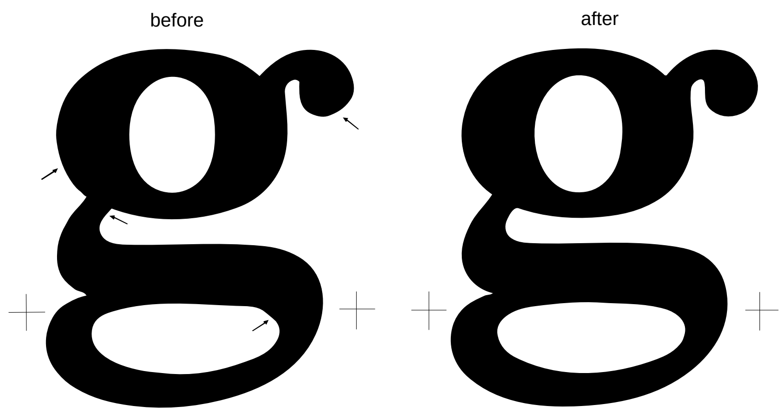


Figure 16. The same letter but with a different curve quality

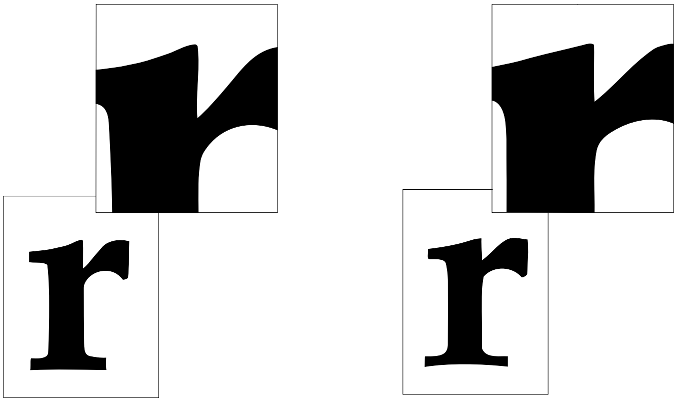


Figure 17. Palatino. Left: detail of the IK-format Right: detail of the PostScript format

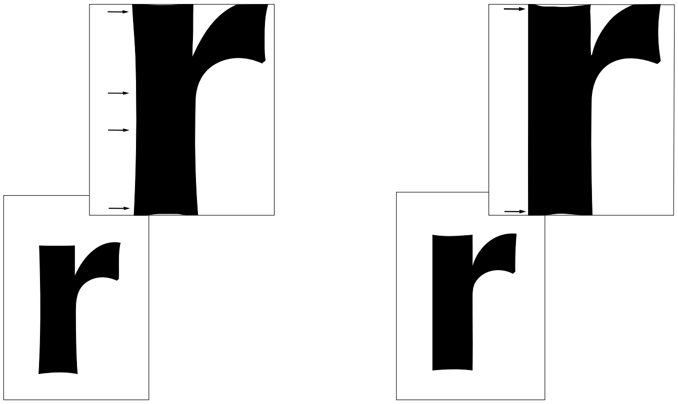


Figure 18. Optima. Left: detail of the IK-format Right: detail of the PostScript format

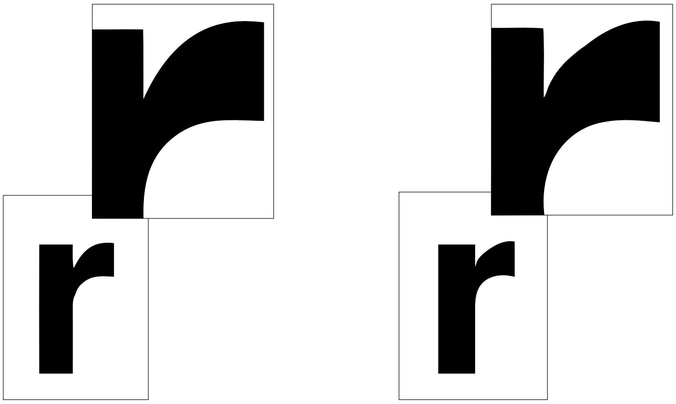


Figure 19. Helvetica. Left: detail of the IK-format Right: detail of the PostScript format

square. Good quality digitizers, drafting machines and film recorders attain a resolution of ± 0.01 mm (corresponding to 2540 lines per inch), which means $15\,000 \times 15\,000$ units per em square. Clearly, in the pursuit of quality, one is not hindered by today's technology.

A high degree of accuracy is necessary not only to satisfy the better-quality printing firms, but also as the most essential prerequisite for desktop publishing (DTP). At first sight this statement seems unbelievable because DTP is mainly used with low-resolution laser printers (at about 300 lines per inch), but it is precisely these printers which employ intelligent font scaling. This is a technique which rasterizes outlines at various point sizes, for low-resolution devices, with the help of hints and markings. However, intelligent scaling can only function properly when the outlines of letters, both individually and collectively, possess a very high degree of conformity and symmetry. In particular :

- For high resolution using film recorders, objective precision of the digital representation must be sufficiently good that outline irregularities can no longer be perceived; that is, typefaces must pass 'optical approval'.
- For low resolution using laser printers, an even greater numerical precision is needed in order to avoid imperfections in rounding; that is, typefaces must pass 'numerical approval'.

We are left with the paradox that for a low-resolution machine a greater precision is needed in the digital representation.

In our quest for accuracy and quality we can now identify some new tasks to be done. These are:

- channel processing
- cutting and pasting serifs
- unitizing
- symmetrizing
- setting extremes
- checking against the rules of digitization.

Each of these is covered, in turn, in the next few subsections.

5.1 Channel processing

Outline typefaces always include straight lines which never run perfectly horizontal or vertical. In their original form, print typefaces were handcrafted. Nowadays, after digitizing — whether by hand or by scanner — these imprecisions are maintained!

With so-called channel processing, lines that are approximately horizontal and vertical are adjusted so that they *are* exactly horizontal and vertical. For the most part, straight lines are defined by a minimum of two support points. Using preset tolerances, programs detect the straight contours of the letter, calculate the mean value of the X and Y coordinates, and set the coordinates of any point detected within the tolerance band to be this mean value (Figure 20).

Unfortunately, you cannot let channel processing run automatically, without supervision. Often, part serifs are required to 'hang' on straight lines, as on the letter 'I', or on curves, as on the letter 'm'. When part serifs are already completed or curves finely balanced, for example, there can be undesirable effects on these features after channel processing.

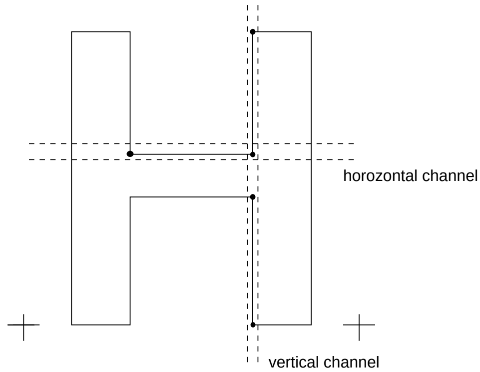


Figure 20. Channel processing in aligning horizontal and vertical lines

5.2 Cutting and pasting serifs

As with all other elements in a typeface, serifs and part serifs are done by hand. Therefore, one cannot numerically digitize an all-purpose serif. Indeed, there are programs for recognizing existing serifs and for cutting them off. After this, one of four possible part serifs is selected, retouched if necessary, and pasted onto the character with the help of mirroring and alignment points (Figure 21).

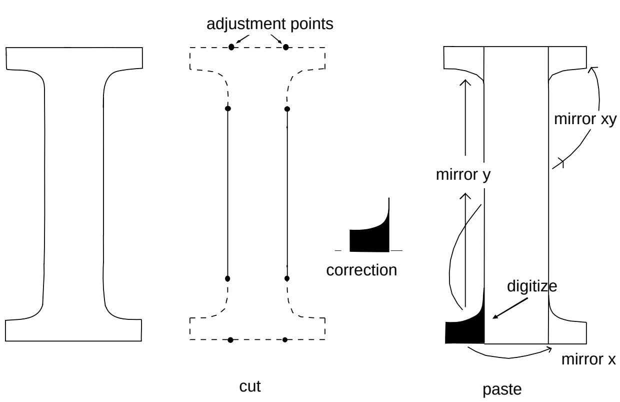


Figure 21. Cutting and pasting serifs

5.3 Unitizing

Sometimes this step can be omitted in typeface production if there are no given values for the character and for its total width. In general, however, there are width lists, which include specifications for left side-bearing, width, and right side-bearing to which our own values have to be adjusted. Special programs execute this action as well (Figure 22). Normally these modifications fall within a range of about ±5% of the width value. The three width measurements can be fitted individually, in pairs, or all together. Only rarely is it necessary to undertake a complete re-drawing, and subsequent re-digitizing, of a letter. Note that the programs use protection zones for the vertical stems, because these features should not be stretched or compressed (Figure 23).

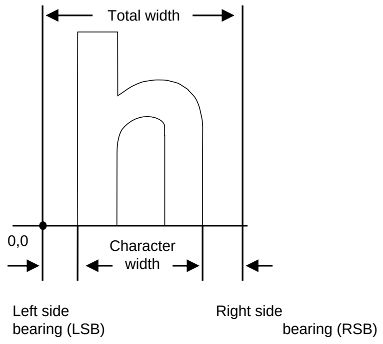


Figure 22. Letter measurements

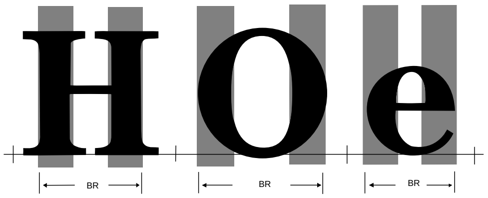


Figure 23. Protection zones to keep certain parts of characters at constant widths; other parts may vary

5.4 Symmetrizing

This step, too, is performed with the help of computer programs. The characters of an alphabet must be placed exactly between, above, or below the typeface lines with full numerical and optical precision (Figure 24).

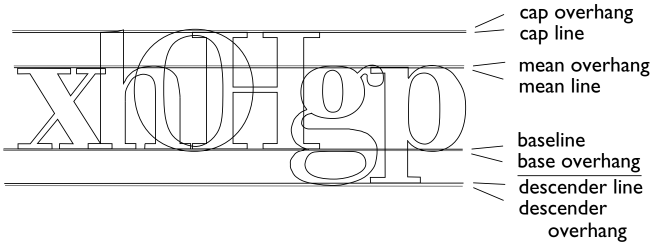


Figure 24. Character lines are guide lines to control heights of characters

Many readers will already know that the letter ‘O’ hangs just a little below the baseline and has a slightly larger cap height than other letters. If left unsymmetrized, the base overhang could receive more, or fewer, pixels than the cap overhang and this, in turn, would interfere with readability.

5.5 Setting the extreme points

We find the extrema of curves at 0°, 90°, 180°, and 270° (see also first section). These are re-positioned on the screen by hand-digitizing or by retouching. There are programs which locate the approximate extrema and automatically set the correct numeric values.

5.6 Testing the rules of digitization

The rules of digitizing are described more fully in the book by Karow [3]. However, before you can employ pattern recognition (automatic hinting) and intelligent scaling, one must test to ensure that the input characters were digitized according to certain rules. Among other things we need to be sure that :

- the black part of the character always lies to the right of the contour direction
- contours do not cross one another
- contours are always closed
- parameters in the ‘header’ for this character, held on computer file, do not contradict the image data for the contours.

6 TYPEFACE DATABASES

Creating a database for typefaces, independent of any particular machine format, is also a part of font technology. As the name implies each of these formats will be dependent on some particular machine. Just as with computers, where the inventiveness of computer engineers and the ever-present pressure from the competition results in a new machine from each manufacturer every three to five years, so also is this true for new typeface formats. To counter this we need to produce typeface data for a general database first, which can then be converted into the ever-changing machine-specific formats. This transformation can occur automatically, under program control, without time-consuming human interference.

Figure 25 shows the production path for a typeface, beginning with the designer’s draft, followed by transfer to a computerized form and then into a different form for some specific typesetting machine. Finally, it reaches the reader, who acquires the ‘information’ via the printed page or a display screen.

There are three ways to enter typefaces into the computer:

- digitizing by hand
- scanning
- interactive design

but in every case modifications to the typeface should be carried out within the machine-independent database. The resolution of the ultimate target machine determines the way in which these data are transformed as well as influencing the format of the ‘digital typefaces’.

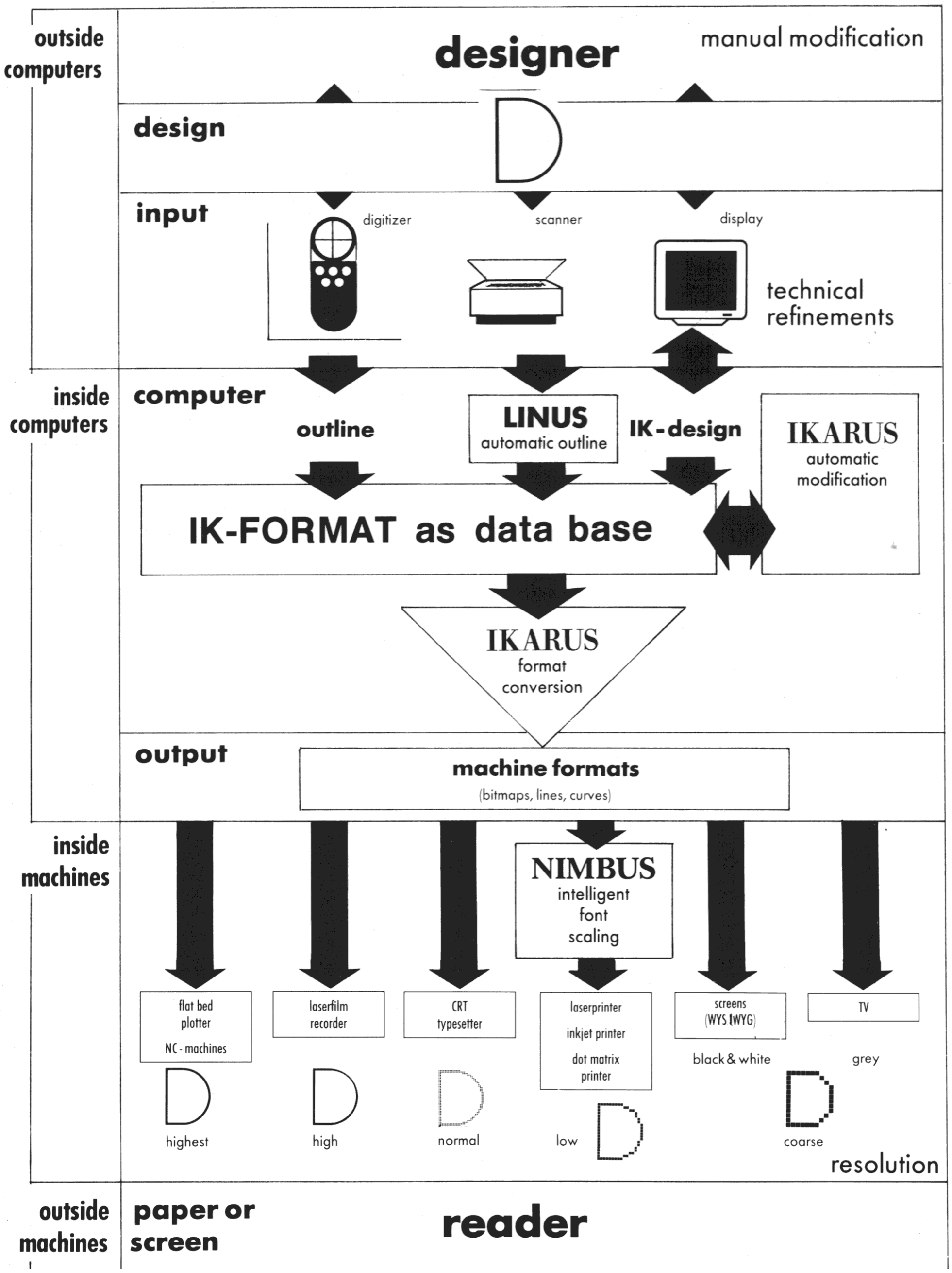


Figure 25. Typefaces on their way from designer to reader; it is important to distinguish the data-base from the machine format

7 CONCLUSIONS

Essentially, typefaces have not undergone any deliberate changes to their appearance as a result of digitization. The object of digitization has always been to obtain a template-true image by computer-controlled reproduction on a drafting machine or phototypesetter. Digitizations of alphabets have had to change, out of necessity, in order to accommodate the new demands of being reproduced on screens and laser printers, but changes have also occurred because of the very possibility of computer-controlled modifications. In our opinion, this has led to improvements in that fonts can now be scaled intelligently (using hints/instructions) and one can interpolate between two or more existing weights of a typeface. This latter technique affords us a far more consistent gradation of weights, in the various versions of an alphabet, than used to be the case in earlier times.

Simplifications of the more complicated details in a font have also occurred. We believe that these changes were necessary, but ultimately they must be seen as compromises, and therefore as changes for the worse. What is meant here, for example, is the straightening of shallow curves in 'Optima' when this font is displayed on a screen, or in small point sizes on a laser printer. Moreover, in photocomposition, corners previously had to be emphasized either by slight over-exaggerations (outer corners) or by subtle indentations (inner corners) and fonts were spaced relatively loosely. This is still reflected in the data for printer fonts but data for display fonts, by contrast, cannot contain these refinements and such fonts need to be more compactly spaced.

All of these factors, taken as a whole, imply that font data have an existence of their own and their computer-controlled images do not necessarily match the original image on the templates. In other words, we have to recognize that there are 'digital fonts' as well as 'fonts in digital format'. Today's typeface manufacturers have evolved from the punch cutters of yesterday into today's graphic designers sitting at their computers. This evolution has been in response to the new technologies of computer-aided design, desktop publishing and office automation. The punch cutter has become a 'digital punch cutter' and this occupation demands additional modifications to the typeface such as interpolating stroke width, italicizing or contouring, as well as generating formats for various systems and reproduction devices. In other words the punch cutter has become a computer aided typography (CAT) specialist.

REFERENCES

1. P. Plass and M. Stone, 'Curve-fitting with piecewise parametric cubics', Imaging Sciences Laboratory, Xerox Palo Alto Research Center (unpublished), March 1983.
2. L. Ruggles, 'Letterform design systems', Technical Report STAN-CS-83-971, Department of Computer Science, Stanford University, Stanford, California. April 1983.
3. Peter Karow, *Digital Formats for Typefaces*, URW-Verlag (ISBN 3-926515-01-5), April 1987.
4. Hermann Zapf and STA, *Hermann Zapf and his Design Philosophy*, Oehms Druck, Frankfurt and Chicago, 1987. ISBN 0-941447-00-6.
5. D. E. Knuth, 'Metafont: A system for alphabet design', Technical Report STAN-CS-79-762, Department of Computer Science, Stanford University, Stanford, California. September 1979.
6. D. E. Knuth, *TEX and Metafont*, American Mathematical Society and Digital Press, Bedford, Massachusetts, 1979.