# A Soft-edged Character Set and its Derivation

### A. J. Wilkes and N. E. Wiseman

University of Cambridge, Computer Laboratory, Corn Exchange Street, Cambridge CB2 3QG, UK

Characters are normally displayed on raster display devices in the form of dot arrays. Humble VDUs often use  $6 \times 10$  arrays spaced on an  $8 \times 11$  grid of positions. Each character is then defined by 60 bits and the entire character set is stored in a read only memory of, say, 96 (characters)  $\times$  6 (bits wide)  $\times$  10 (bits high). This is 5760 bits all together and accounts for only a tiny proportion of the component cost of the unit electronics. In some of the 'better' terminals more characters may be provided. Some character sets may be held in random access memory and thus be loaded dynamically from a user's definitions, but it remains the case that the character generator logic is relatively small compared with everything else. Indeed, this is increasingly true with more and more upmarket displays—most of the increased function being devoted to picture storage and manipulation. A binary dot array is not in fact the best way to render a letterform on a CRT display. The dottiness produces a pretty crude and unsatisfactory image which is ugly and often hard to read. Raster displays do, it is true, tend always to be seen showing dotty and ugly pictures but diagrams are not intended for 'reading' in quite the same manner as running text. Fortunately, this dottiness is not intrinsic and solutions are known for arbitrary images, including text. Quite stunning improvements can be achieved by simple methods which can be cheaply implemented for character displays. This paper discusses the techniques and presents a complete alphabet definition as an example of the method in use.

### INTRODUCTION

A picture generated for output on a device of limited resolution ought to be processed to exploit the device characteristics as well as possible so that the distortion and degradation are controlled and minimized. Except for stylized pictures tailored to the display properties, the only freedom we have is in the way the display is controlled to reproduce the given picture. The computation for a raster display is in fact a sampling of the image at a spatial frequency given by the pixel spacing. Consider the edge of some feature in the picture where differently shaded polygons meet (see Fig. 1).

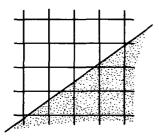


Figure 1. A feature boundary intersecting the pixel grid.

This edge intersects the pixel grid, which is the spatial sampling frequency, and at each sample (pixel) we must compute an appropriate shade. Except when the edge aligns perfectly with the grid, this shade will be neither that of one polygon or the other but an intermediate value fixed by the amount of each which falls within the pixel boundary. Suppose that the edge is black on one side and white on the other, and that 4 bits are available to specify a shade encoded as a binary value between zero and 15. The situation shown above might then give rise to the following pixel values (see Fig. 2).

This use of grey scale to form what is intended as

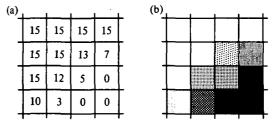


Figure 2. (a) Grey values in vicinity of boundary. (b) Resulting image.

simply a black-white edge works extremely well when the pixel size is reasonably small. The subjective effect is in fact of a hard straight edge rather than of the soft defocussed edge which one might expect. The reason for this has to do with the way in which the brain appears to seek and accentuate line features using global clues as well as local detail. If no grey scale is available the actual pixel values will of course be zero when the wanted value is below eight, and one otherwise which produces the jagged edge which is the familiar and objectionable effect we are trying to rid ourselves of here (see Fig. 3).

The jaggedness obviously will vary in detail as the edge moves over the pixel grid. A given edge may in fact show itself in many different forms, called aliases, and a slowly moving image produces edges which shimmer as a result of the succession of different aliases which the

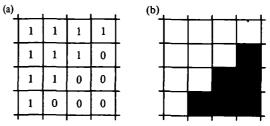


Figure 3. (a) Binary values in vicinity of boundary. (b) Resulting image.

movement induces. The use of grey scale in the manner suggested above is called anti-aliasing; the grey values may be computed in several different ways. The basic idea comes from sampling theory, which states that with a sampling frequency of f, accurate reconstruction may be achieved for images containing no components above f/2. Put simply, this means that if an image is filtered so that such high frequencies are removed, then the jaggies will go away. All methods for anti-aliasing amount to filtering the image in this way, but the methods differ in their accuracy and the amount of computing required. Quite simple, quick algorithms are often good enough, although rendering movement satisfactorily is difficult. For the present purpose a simple method will be used, based on pixel averaging. The letterforms will be specified to a high resolution so that they can be mapped onto the pixel grid by amalgamating squares. The total of black squares per pixel is then counted and used to emit a corresponding grey value for each pixel.

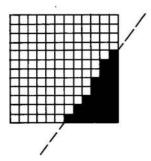


Figure 4. Super-sampling a pixel to determine its grey value.

Shown in Fig. 4 is a single pixel divided into a binary array of dots, presumed fine enough to allow a count of dots to be a sufficiently accurate indicator of the grey value needed. Letterforms when designed will usually be to an extremely high resolution so this condition is easily met.

In summary, the derivation of an alphabet to a low spatial resolution from binary high resolution masters can use grey values to restore some of the lost detail and fool the eye into hardening and smoothing the image contours. Shown in Figs 5-7 are some examples of the sort of results we have obtained starting with a particular alphabet and deriving others from it.

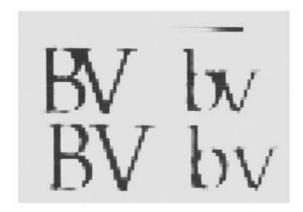


Figure 5. Enlarged letters, 20 pixels high, 2 bits per pixel.



Figure 6. Enlarged letters, 56 pixels high, 4 bits per pixel.

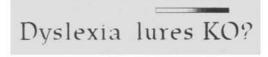


Figure 7. Normal sized letters, 56 pixels high, 4 bits per pixel.

### **IMPLEMENTATION**

Experiments in letter design and text processing for print quality applications have been going on in this department for some years.3-5 Alphabet definitions associated with this work were available to us in the form of high resolution polygonal outlines. Each letter is formed from the superimposition of a few such polygons, comprising up to a hundred edges or so in a coordinate space delimited by 16 bit integers. A program to scale and scan convert these letterforms into a run-encoded form for writing text to precision raster plotters and film setters was used to preprocess the letters for the use described here. Thus our input was a run-encoded binary array for each letter on a grid roughly 700 x 700. The pixel averaging algorithm we used is shown below.

```
(LET pixels = VEC width - 1)
                          // Accumulated values,
                          // which start off at zero
   FOR i = 0 TO width -1 DO pixels!i = 0
                          // Read in N lines of pixel
   FOR line = 1 \text{ TO } N
                          // runs and accumulate them
   DO $(
          JNTIL end_of_line
        DO $( LET first = READN ()
                          // Read in a span of blacks
                LET last = READN()
                FOR x = first TO last
                DO pixels! (x/N) + = 1
             $)
       $)
   // Write out a line
   FOR x = 0 TO width -1 DO write(pixels!x)
   newline ()
```

### \$) REPEATUNTIL end of data

The output consists of rows of width pixel values, the whole array being terminated by a value of -1. In practice; some compression is used: leading and trailing zeroes are suppressed and each row is preceded by an offset and the number of non zero data values on it.

The pixel values will range between 0 and  $N^2$  and will need to be mapped to the actual set of grey values which the output device can handle. The choice of mapping is up to us. We can make use of the opportunity to tune up the performance of the overall method to compensate for brightness non-linearities in the display, filtering faults and the human vision process. If the output device has more grey values than the filter emits, then the mapping can be done by a table of outvalues indexed by the invalues. If, on the other hand, the output device has fewer grey values than the filter emits—as was the case for our experiments—the mapping can be done by a binary search in a table of invalues. The search code is:

```
outvalue := 0
increment := N<sup>2</sup>/2
UNTIL (increment = 0)
DO $(

IF (invalue > maptable!(outvalue + increment))
THEN outvalue + := increment increment := increment/2
$)
```

The vector maptable records the invalues at breakpoints in the mapping. The outvalue deduced by the algorithm is the index to the next lower invalue to the one given. It evidently requires that the mapping is monotonic increasing. The filtered and remapped data for each letterform can now be stored in a pixel array, say width pixels wide, height pixels high and depth bits deep. An alphabet in this form for width = 19, height = 21 and depth = 4 is reproduced in the Appendix. The mapping chosen for this example is linear—that is the invalues were arranged in 16 equal intervals, each corresponding with a particular outvalue.

### REMAINDERING

In the latter case, where there are more invalues than outvalues, a cumulative quantization error is introduced, causing the integrated grey level of the resulting picture to be too low by an amount determined by the image

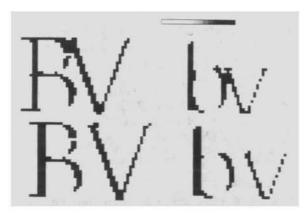


Figure 8. Enlarged letters, 30 pixels high, 1 bit per pixel, no remaindering.

details. Each consultation of maptable gives an error (or remainder) of

invalue - maptable!outvalue

which is positive or zero. The cumulative effects can be

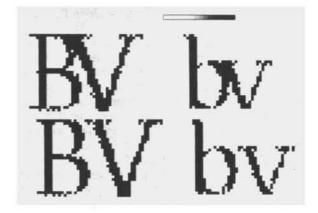


Figure 9. Enlarged letters, 30 pixels high, 1 bit per pixel, remaindered.

minimized by distributing the error over neighbouring pixels, rather than simply throwing it away. The simple technique of adding it to the pixel value to its right works quite well, as Figs 8 and 9 show.

Remaindering in both coordinates is theoretically better but probably unnecessary when several levels of grey are available for the output. As might be expected, the effect is most noticeable when the size of the error terms is large: that is, there are few grey levels available in the output. With 16 levels, we could not easily distinguish the effects of remaindering for the alphabets we wished to display on a Sigma 5684 terminal.

## USING THE DATA

The use of anti-aliased characters on displays which have a grey scale capability is relatively straightforward. A line of text is scanned height times, constructing and emitting on each scan the corresponding pixels for each letter required in turn. The basis of the method is shown below. A vector txt holds the line of text being processed, with its first word indicating the number of characters. Each character is represented by the address where its pixel array starts; a pixel array is a vector of height rows each held as a vector of width pixels.

```
FOR i = 1 TO height
$( scanptr := 1
FOR j = 1 TO txt!0
$( letter := text ! j
FOR k = 1 TO width
DO $( scanline!scanptr := (letter!i) ! k
scanptr + := 1
$)
$)
$scanline!0 := scanptr
emit( scanline )
$)
```

A piece of logic to do the same thing is simple and would be a quite insignificant addition to the hardware of a character based display. The pixel arrays are held in ROM, depth bits deep instead of one, and the depth bits shifted to the video generator pass through a Digital to Analogue Converter (DAC) en route.

The row counter (height bits) and character code address the ROM in the usual way causing the emission of width × depth bits (instead of width). These pass into depth shift registers whose outputs are connected to the

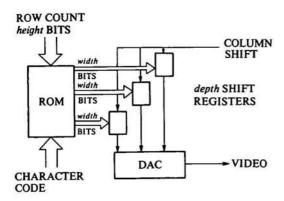


Figure 10. Basic logic for a soft-edged character generator.

DAC which emits video. No timings are altered and the circuits are not expensive.

### INTERLACING

With traditional simple VDUs, the use of interlacing does not greatly increase the overall legibility of the character set, which will usually have been designed on a simple dot matrix pattern. Doubling the resolution allows a few refinements, but no major improvements are possible: the image is still locked to the pixel grid. When using a grey level display, however, advantage can be taken of any additional spatial resolution: the increased number of pixels can be immediately taken advantage of (no laborious tweaking by hand is necessary), and will be reflected directly in the quality of the final image. The example character set in the Appendix was sized to allow display of 24 lines of good quality text on a 512 line interlaced screen.

### **FURTHER USES**

This exchange of grey values for spatial resolution in images as perceived is a reversible effect—that is, we can also use binary pictures to create a grey scale effect by losing spatial resolution (this is the means by which continuous tone images are turned into halftone images for printing by ink transfer). It seems very likely that a digital film setter such as is used to make page images for printing could exploit soft edging in the way described in this paper to remove the bumpiness in letters of which the industry constantly complains. The additional apparent spatial resolution which soft edging produces should still be there after the image is thresholded by the film emulsion, provided that image integration prior to exposure is well chosen. We hope to be able to verify this soon.

### CONCLUSIONS

We have presented a simple technique for anti-aliasing characters and demonstrated the benefits in picture quality which can be easily achieved. As better displays become available, we think some such technique should be implemented in hardware. Suitable enhancements to handle variable widths and heights are not difficult to devise and, indeed, the example character set given below is intended for use with variable pitch.

By fixing the size in relation to the display resolution and accepting less freedom in the letterforms a soft edged character set could be designed which would reproduce even better than the examples shown here. Graphics arts designers might not find the restrictions of form fully acceptable, but we believe that the raster graphics medium deserves their attention.

## REFERENCES

- F. C. Crow, The aliasing problem in computer generated shaded images. Communications of the ACM 20, (1977), pp. 799–805.
- J. E. Warnock, The display of characters using grey level sample arrays, Xerox Palo Alto Research Center Research Note CSL-80-6, (1980).
- A. M. Pringle, P. Robinson and N. E. Wiseman, Aspects of quality in the design and production of text. ACM SIGGRAPH 13, (1979), pp. 63–70.
- N. E. Wiseman, C. I. C. Campbell and J. Harradine, On making graphic arts quality output by computer. *Computer Journal* 21, (1978), pp. 2–6.
- N. E. Wiseman, Computer designed letters. Information Design Journal 1, (1980), pp. 218–222.

Received May 1981

© Heyden & Son Ltd, 1982

### APPENDIX

The example is based on an alphabet designed by David Kindersley in digital form using the ELF design system. The data is presented as rows of hexadecimal numbers, each one describing one line of a character. The data for each character is prefixed by \$\chicknothernormalcolor \chicknothernormalcolor \chicknothernorma

Figure 11. A screen reproduction of the alphabet drawn at actual size.



# **ALPHABET LISTING IN HEXADECIMAL**

\$! <b>/</b> 41	\$* #42	\$# ##3	\$\$ \$44	\$\$ #45
	42.52			
	1F14E		••••••	
1	77.A5 16.26		•••••	87
<del>59</del> F3			15.6	2537B 61.833D1
EFE		17 157	27EAF93	A 44.1C5
CFC	***************************************	8C1D7	9827.BAA1 .3F337.9	671499 15C1
8F9	***************************************	13F58F211	.3F937.9	103
6F7 4F5		157CF8FF661		7A
2F4		1E93F5111 16AFCDF8861	27.A6FB	3D1.3358
D3		5F19C	.24628.9.49	98A72
		9C.C9	7EFE5C35	5B14733
	***************************************	****************	25.7	. 31
•••••		***************************************	••••••	
\$+ #53	\$. #54	<b>\$- #</b> 55	\$. #56	4/ 457
		4- 122	***************************************	\$/ #57
	***************************************	***************	***************************************	
		**************	••••	
33	***************************************	***************************************	***************************************	91
D7				79
D7	***************************************	***************************************		E2
D7 4777FE7773	***************************************	••••••		1D2
D7		***************************************	•••••	88
D7	***************************************	3566551	•••••	97
D7			***************************************	97
c5	33 E5	***************	33 F3	2D1
	.1E1	•••••	1	
	. 45	•••••	***************************************	
		***************************************	•••••	
\$5 #65	\$6 #66	\$7 #67	\$8 #70	\$9 #71
	***************************************			
	***************************************			
1888951	1353	.379898BB3	1	448EB2
4F2 5F1	85	1EB	1F3.6F6 6F1F9	983EC
7D	2F2	8F3	7F2E8	.7f4F6
8D4971	TD 25A93 BF5.2CE4	9F2 2F9	4F9.2E2	.8F16F9 .6F58F6
1FA	FF 1FB	AF1	774DE3	. 1ED 1FF 3
9E	FFBE FC9E	4F7	6D2EB	3CE759F
AB	CDAB	5F6	CBAD	E4
3E5	8F41E4 1CE7365	DD	AF1C8 3FB35A	96
453	342	.2	2651	12
****************		****************	****************	
•••••	***************************************	***************************************	***************************************	
\$? #77	\$@ #100	\$A #101	\$B #102	\$C ∉103
		***************************************	***************************************	
***************************************	***************************************	***************************************	***************************************	
	4247871	11	112111	21.1
36B81	343C6	FF2,	7DF78ED2	4369EBC
1A1.2DC	451C5 284D	5FF7	9E2FB	3A11BF
C93F1	92254B7C4	F56F3	9E9D	7CA
8E179	CB796 .2B487468	5E11F8	9EB9 9F44781	c9
57	.3A77B176 .4B5BD83	F55F4 5F8788F9	CFCBBF81	.1F7
B6	.1EC616E4.27	AAAD	9EEC	CD
B2	D2241.274 6B	F45F4 5E1E9	9EBC 9ED7	7F534
1E3	A9	13ED33DF4	3EF321691	2CE721247
1	68611 47542	1463345441	4655752	37643
	***************************************		***************************************	
		***************************************		***************************************
\$I #111	\$J #112	\$K #113	SL #114	\$H #115
***************************************	***************************************	***************************************	***************************************	
****************	***************************************		***************************************	
2121	***************************************	•••••	***************************************	
49FB5	4AF D6 2F6	.18FF81.6CF83 CB2D2	5BFB4	.19CFC3FF96 8FF47DF3
3F5	2F6	CB1C4	5F3	99F9C8F3
3F5	2F6	CBA7	5F3 5F3	89E13D3F4 B74F5882F5
3F5	2F6	CB4D1	5F3	B61DBD31F6
3F5	2F6	CB6F6	5F3	C6.8E2.3D.1F6 D4.3F7.97F7
3F5	2F6	CBBE2	5F3	E31.CC.E2E8
3F5	2F5	CB1CD1	5F3	F36F7CD9 F2ZEF7D9
27F73 25543	. 2FC 4A5	4FD417FC41 4644146441	2AF5444C	.26F52BF2.14FD4
***************************************	143	*****************	35454444	. 34542 35 5544
***************************************	***************************************	***************************************	***************************************	
		/		

# Downloaded from https://academic.oup.com/comjnl/article/25/1/140/527310 by guest on 09 April 2024

### A SOFT-EDGED CHARACTER SET AND ITS DERIVATION

\$å #46	\$' #47	\$( #50	\$) #51	\$* #52
***************************************	42	***************************************	***************************************	
	77	49	4C4	
66AE9		1D1	AE	
9A8F4 F83F6	***************************************	97	1F7	3E72B2
FB3F3		4E	6F1	2FB.5D1
8F6.89		7B		46FFF 56653
2FF 6.AF 8		9B	1F3	CEF 6
67.AF76E		7E	3E1	6C.DE2
F68FF .1F91FF91	***************************************	4F3	6B	1E3.4FA
AF7323617E9		8E3	2c	
35532		AE3	291	
***************************************				
			***************************************	
\$0 #60	\$1 #61	\$2 #62	\$3 #63	\$4 #64
		•••••	***************************************	
	•••••			
2	***************************************	***************		
536BEA2 3B3EE1	7F2	548DD5 B6BF3	182	FF
1E33FA	.8FF 2	.231FA	2A1	3DDC
8CAE2	2E9F1	CB	2A1	D4CC
. 1F62F7	6F2	3E1	18F6	3D1.CC
.2F62F6 .1F92F4	6F2	1A3	AF1,	.,D4CC
DD5E1 8F4A9	6F2	1B2	3F5	199888FF 62
1DE24C	6F2	1B2	4F2	BC
2DE62137		.2E8676764	.7EA3481	BB
***************************************	***************************************			
	***************************************	***************************************		
\$: #72	\$; #73	\$< #74	\$= #75	\$> #76
	***************		**********	
***************	*****************	*****************	******************	
•••••	•••••	74	***************************************	
	*******************	79	***************************************	.1AA
***************************************	***************	5B		6FA
	****************	.1D2	. 158888885	6FA
	******************	.C4 6F6	***************************************	6FA B7
31	22	.BE5	. 168888786	87
4F	E5	DE3		87
51	33 E5	3FB1	***************************************	87
1	. 1E1	2	***************************************	1
	. 45	*****************	***************************************	
	***************************************			
\$D #104	\$E #105	\$F #106	\$G #107	\$H #110
*******************		***************************************	••••••	
	***************************************			
2	***************************************	***************************************	21.1	233111211
.19EF667BF81	8DF988D3 9E73	5F337	4379EBC 3A11BF	9FF827FF6 BCE9
CB6F7	9E1	5F31	1F2,1F	BCE9
CB DE CB 8F2	9E	5F3 5F3	8CA	BCE9
CB5F4	9F5543	5F3	. 1F6	BF455455FA
CB4F4 CB5F2	AF 4332	5FB887	. 2F6 . 1F9888	CF444344F9
CBBC	9E	5F3 5F3	9E 8F49F	BCE9
CB7A	9E7	5F3	1ED29E	BCE9
.,4FD3211451 .,5557662	3DF3#44B 455#4443	29F62	20E6213A3	4FE414F83
.,	******************	******************	***************************************	
	***************************************	***************************************	***************************************	
SN #116	\$0 #117	\$P #120	\$Q #121	\$R #122
*******************	***************************************			
	***************************************	***************************************		
***************************************	***************************************	***************		
.38FF558FC5,	436AEB2	11	4379FB2	11 BDF67CE4
BF01£5	3B3DE3 1E32EC	5F34F9	3A13DE2 1F22FC	AD1ED AD9F1
B7AE2E5	8B9F3	5F3DC	8C9F2	AD7F
872EAE5 87.7F4E5	D83F7 F71F8	5F3E6 5F31176	D84F6 .1F61F8	ADAA
87.1CDES	. 1F7F8	5F963	.2F61F7	CFEEE7
874F7.E5	F91F6 CD4F2	5F3	. 1F9 1F5 DD 4F1	AD8F2
872F9E4 878FF4	8F49A	5F3	8F*3.A9	AD8F2
.14FA4DF5	1ED22D1 2DE63.361	5F3	tED26F9D1 2DE6219F7	AD1EA 3EF418F53
. 14634 252	3763	35533	463.8F4	46441752
			***************************************	
		***************************************		

# Alphabet Listing in Hexadecimal-cont.

\$S #123	\$T #124	\$U #125	\$¥ #126	\$W #127
			***************************************	
			••••••	•••••
	11	***************************************		•••••••
448BF5	.8D89AFC89D4 .546F364	.5AFB66AFC5 4F44F4	38FF76BF85	29FF73F96BF84 AC8FE7A
7C53	.431.5F353	.,4F44F4	3F5D3	7E1D8F4A7
BD1	5F3	.,4F44F4	DA3D 7F188	3F42E1F8D4 F77A.AD1F1
58BFF6	5F3	.,4F44F4 .,4F44F4	2F6D3 CB #C	CAC5.6F2.5C 9D.1F1.1F7.89
2EE	5F3	.,4F44F4 .,4F54F3	6F2.98 1F71D3	6F16BBD.A7 3F4A77F2D4
9F 47E	5F3	2F97D	BC4D	E9E22F9E1
9A7 F863349	5F3 29F63	9F51D3 AF94373	5FC8	BFCDFC
23#43	26443,	2431	6D	292284
***************************************		***************************************	***************************************	
\$1 #135	\$? #136	s_ #137	\$` #140	\$a #141
**************	*************			
267A4		***************************************		
D7		***************************************	182	
	92 6FB1	***************************************	. 2A92	
D7	. 45E2B . 27 . E. 49	34	***************************************	
D7	. 9 Σ 58	52	••••••	2794
	51E63 D1		***************************************	291.8F3 62 b9
D7	D1	172	• • • • • • • • • • • • • • • • • • • •	DC
	D1	5	***************************************	68FF,
D7	D1	*************	*************	D64FA
D7 277E3	••••••			442.242
		***************************************		
•••••	***************************************	***************************************		
\$g #147	\$h #150	\$1 #151	\$j #152	\$k #153
	***************************************	***************************************	***************************************	
		***************************************	***************************************	***************************************
		***************************************	***************************************	
	£6	3	12	6£5
	E6	55	82	E6
149844	E6287	397	773	E6.59A2
84.2BFC	EE 14E8	8C	F5	E6.2C1 E6.B4
A8EC	F75F1	8c	F5	E669
C8AA BAFF	E64F1	8c	F5	E63F6
7E3FD CD426EB	E64F1 2F9117F4	8C	F5	E6.6E4
351.AA	35411544	2533	F5	3541.4541
4A7 4F62C1	***************************************	***************************************		
5DA441		•••••	5083	
\$q #161	\$r #162	\$s #163	\$t #164	\$u #165
*************			*************	
		***************************************		
			*************	
	***************************************	••••••	*************	
			72	
2695#	. 189 1287	48BA	2F52	. 187 1 . 476
A3.7FF	1F9842	2A6B 9A12	7F 85 1	4F1AA
A8AF	5F	8F953	2F3	4F 1 AA
BA9F	5f	BC	2F 3	4F2FB
7F1FF	5F	37B D743A2	2F 4	1F7FB 9E55DB2
4416E	1543	4433	14	33.333
6E	•••••			***************************************
2CF7			•••••	***************************************
\$E #173	\$   #174	\$) #175	\$ <sup>7</sup> #176	
	***************************************		***************************************	
121	***************************************	4891	***************************************	
58 2D1	2B	3F5	***************************************	
89	5E	E7	***************************************	
¢6	5E	C9	***************************************	
2£1	5E	3F92	23	
1CB2	5E	381	5A9E1	
cc	5£	69	23F54A BC11A2	
5F	5E	1F2	1973	
4F2 2F4	5E	89	***************************************	
081	1	182	************	
*****************	***************************************	**************	***************************************	
•••••	***************************************	***************************************	***************************************	

# Downloaded from https://academic.oup.com/comjnl/article/25/1/140/527310 by guest on 09 April 2024

### A SOFT-EDGED CHARACTER SET AND ITS DERIVATION

\$X #130	\$Y #131	\$Z #132	\$[ #133	\$\ #134
***************************************	***************************************	***************************************	***************************************	***************************************
	***************************************	***************************************	***************************************	
****************			***************************************	
***************************************	***************************************	***************************************	6A75	
.49FF67DF84	29FF87BF94	1	£6	.4D2
6F51E3	4F7D4	9SF9	E6	. 3F8
BE1A8	9E389	91ED1	E6	CE
2EA.5C1	EC.3D	1AF4	E6	6F5
6F6E3	4F7D3	4FA	E6	FB
BF9	9FB	EDS	E6	9F2
7FB	2F6	BF6	<u>E</u> 6	3F8
1E9F5	2F6	3FB	E6	CE
A71AE2 5C 2EB	2F6 2F6	8F682	E6,	1EB
1E26F6	2F6	3FB82	E5	9F2
.34EA22FF43	16F93	CF9767676F2	E6	3F7
.3554135543	24643	1211212113	E6	1
***************************************		***************************************	B985	
***************************************	•••••	***************************************	***************************************	
•••••		***************************************	***************************************	
\$b #142	\$c #143	\$d #144	\$e #145	\$£ #146
***************************************		***************************************	***************************************	
***************************************		***************************************	***************************************	
***************************************	***************************************	***************************************	***************************************	
•••••		***************************************	***************************************	
8E3		2AD	***************************************	1
1F4		6E	****************	1AAE3
1F4		3	***************************************	99.2
1F4		6E	***************************************	B9
1F43882	138A9	269AE	14971	CB22
1FE. 1CD1	7598	A3.7FF	84.2EA	2FE55
1F9 2F6	5828	6BFF	5B7F1 BE778AC1	B9
1F4C9				B9
1F787	BA	BA9F	BA	B9
1FE 1E2	7F1	7F1FF	7E2	89
1FF 8275	CD4242	1DB43BE3	CD4242	2EB2
1252	352	44.143	352	2542
		***************************************		
***************************************		***************************************	***************************************	
***************************************		***************************************	***************************************	***************************************
\$1 #154	\$m #155	\$n ∮156	\$o #157	\$p #160
*****************		***************************************	*************	
	••••••	•••••	***************************************	
	••••••	•••••••		
	•••••••••••	•••••••		
	•••••••••••	•••••••		
388				
388				
3BB	452163361,		1971	7822882
388			1971 84.28E1	7822882
3BB				.7822882 .1FB11CD1 .1F92F6.
3Bb				
3BB				
3BB	. 452163361	.6833871	1971 .84.28E1 .5B. 2E8 .80. AC .C6. 8B. .8A. A9 .TE2.1D3.	.7822822 .1FB11CD1 .1F9.2F6. .1F4CA .1F4AA .2F7C6.
38B 8C				
3BB	. 452163. 361. .2FB3TF672BD1. .FD. BF2.2F7. .FT. 7FC9. .F5. 5FB9. .F5. 5FB9. .F5. 5FB9. .F5. 5FB9.	.6833871 .ED22CC .FF . UP4 .F7 . F6 .E6 . E6 .E6 . E6 .E6 . E6 .E7 9112F82 .5502.0851	18971 .84.28E1 .58.2E8 .A8.AC .C8.8B. .BA.A9 .TE2.1D3 .CD8264	
38B 8c				
388	. 452163. 361. .2FB3TF672BD1. .FD. BF2.2F7. .FT. 7FC9. .F5. 5FB9. .F5. 5FB9. .F5. 5FB9. .F5. 5FB9.	.6833871 .ED22CC .FF . UP4 .F7 . F6 .E6 . E6 .E6 . E6 .E6 . E6 .E7 9112F82 .5502.0851	18971 .84.28E1 .58.2E8 .A8.AC .C8.8B. .BA.A9 .TE2.1D3 .CD8264	
388	. 452163. 361 . 2FB3TF6728D1. . FD. 8F2. 2F7 . F7 . 7F . C9 . F5 . 5F . 89 . 68733897 . 6475	.6833871 .ED22CC .FF . #F4 .F7 . F6 .E6 . E6 . .E6 . E6 . .E6 . E6 . .E7 9112F82 .532_4541	14971 .84.28E1 .58.2E8 .A8AC .C8.8B .BA.A9 .T22.1D3 .CD4264 .352	
38B				
388		.6833871 .ED22CC .FF. 4F4 .F7. F6 .E6. E6. .E6. E6. .E6. E6. .E79112F82 .5542.4541	1971	
38B			1971 84 . 28E . 58 288 . 86 AC . 60 88 . 88	
388		.6833871 .ED22CC .FF. 4F4 .F7. F6 .E6. E6. .E6. E6. .E6. E6. .E79112F82 .5542.4541	1971	
38B		\$2.591.00000000000000000000000000000000000	1971	
38B			1971 84 . 28E . 58 288 . 86 AC . 60 88 . 88	
38B	#\$2163361 .2FB3TF6728D1 .FDBF2.2F7 .F7TFC9 .F5SFB9 .F5SFB9 .F5SFB9 .6B733897.6A75.		1971	
38B	#\$2163361 .2FB3fF672BD1 .FDBF2.2F7 .F7.1FC9 .F5.5FB9 .F5.5FB9 .F5.5FB9 .6873B97.6A75.	\$833871 ED22CC FF, 4F4 F7, F66 E6 E6 E6 E6 E6 E6 E79112F82 5592.4591	14971	7822882 .1FB11CD1 .1F9.2F6. .1F4CA. .1F4AA. .2F1C6. .2F01E2 .1FF7374 .1F652 .1F4. .1F4. .9F92
3BB	#\$2163361 .27837F6728D1 .FD8F2.2F7 .F77FC9 .F55F89 .F55F89 .F55F89 .68733897.6A75.		18971 18971 84 28E1 55. 2268 88 88 88 99 722 1D3. CD4264 352.	
38B	#\$2163. 361 27B3fF672BD1 .FD. BF2. 277 .F7. 7FC9 .F5. 5FB9 .F5. 5FB9 .F5. 5FB9 .6873B97. 6A75.	\$2.00	11971	7822882 
3BB	#\$2163361 .27837F6728D1 .FD8F2.2F7 .F77FC9 .F55F89 .F55F89 .F55F89 .68733897.6A75		18971	7822882 
38B	#\$2163. 361.  2FB3TF672BD1  FD. 8F2. 2F7  F7. 7F. C9  F5. 5F. 89  F5. 5F. 89  F5. 5F. 89  6873897. 6A75.	\$x \$170	\$\frac{14971}{\tag{2}}\$. \tag{84.28E1}{\tag{2}}\$. \tag{84.28E1}{\tag{2}}\$. \tag{85.288}{\tag{86.48}}\$. \tag{86.48}{\tag{86.48}}\$. \tag{89.49}{\tag{2}}\$. \tag{20.28E2}{\tag{2}}\$. \tag{20.28E2}{\tag	78228821FB11CD11F9.2F61F4AA2F1C62F01E21FF73741F6521F41F49F92.  \$z \$172
3BB	#\$2163361 .27837F6728D1 .FD8F2.2F7 .F77FC9 .F55F89 .F55F89 .F55F89 .68733897.6A75		14971 14971 84 .28E \cdot	7822882 
3BB	#\$2163361 .2FB3TF672BD1 .FDBF2.2F7 .F7TFC9 .F5SFB9 .F5SFB9 .F5SFB9 .6B733897.6A75 \$447139A4 .891EE2B5 .5CS8C6B2 .2FA37BD. .E31C.3E13A	\$2. \$170  \$2. \$170  \$2. \$170  \$3. \$170  \$3. \$170  \$4. \$170  \$5. \$6. \$6. \$6. \$6. \$6. \$6. \$6. \$6. \$6. \$6	\$\frac{14971}{\tag{2}}\$. \tag{84.28E1}{\tag{2}}\$. \tag{84.28E1}{\tag{2}}\$. \tag{85.288}{\tag{86.48}}\$. \tag{86.48}{\tag{86.48}}\$. \tag{89.49}{\tag{2}}\$. \tag{20.28E2}{\tag{2}}\$. \tag{20.28E2}{\tag	7822882 .7822882 .1FB11CD1 .1F9. 2F6. .1F4. AA. .2F7. C6. .2FD. 1E2. .1F7374 .1F652. .1F4. .9F92. \$z \$172
38B	#\$2163. 361.  2FB3TF672BD1  FD. 8F2. 2F7  F7. 7F. C9  F5. 5F. 89  F5. 5F. 89  F5. 5F. 89  6873897. 6A75.   \$4 \$167	\$x \$170  \$x \$170  \$x \$202  \$x	\$\frac{14971}{\tag{2}}\$ \$\text{84.28E1}{\tag{2}}\$ \$\text{58.288}{\tag{8.48}}\$ \$\tag{86.48}{\tag{4}}\$ \$\tag{2}\$ \$\text{50.288}{\tag{2}}\$ \$\tag{2}\$ \$\text{50.288}{\tag{2}}\$ \$\text{50.288}{\tag{2}}\$ \$\text{50.288}{\tag{2}}\$ \$\text{50.84}{\tag{2}}\$	7822827821827821810179276174AA271652701621774176521741749792.  \$z \$172
38B	**************************************	\$233871 ED22CC FF 4F4 F7 F6 E6 E6 E6 E6 E7 112F82 3542, 4541 \$2 \$170 \$2 \$170 \$3 \$170	\$y \$171  \$y	7822882 .7822882 .1FB11CD1 .1F92F6 .1F4AA .2F7C6 .2FD1E2 .1FF7374 .1F652 .1F4 .9F92 \$z \$172 \$z \$172
\$v \$166	#\$2163. 361.  2FB3TF672BD1  FD. BF2. 2F7  F7. FF. C9  F5. 5F. B9  F5. 5F. B9  6B73897. 6A75.   \$4 \$167   \$4 \$167    5AA*. A7. 39A*.  89. 1EE2. 85  5C. 58C6. B2  2F. A37B. D.  E31C. 3E13A.  A749. 0567.  7A85. 8A9*.	\$x \$170  \$x \$170  \$x \$202  \$x	\$\frac{14971}{84, 2861}\$ \$\frac{84}{8}\$ \$\frac{286}{8}\$ \$\frac{8}{8}\$ \$\frac{8}{8}\$ \$\frac{8}{8}\$ \$\frac{8}{8}\$ \$\frac{8}{8}\$ \$\frac{8}{8}\$ \$\frac{8}{8}\$ \$\frac{8}{9}\$ \$\frac{171}{8}\$ \$\$\frac{4}{8}\$ \$\frac{1}{9}\$ \$\frac{8}{8}\$ \$\frac{1}{9}\$ \$\frac{8}{8}\$ \$\frac{1}{9}\$ \$\frac{8}{8}\$ \$\frac{1}{8}\$	7822882. 1FB11CD1 1F9.2F6. 1F4.CA. 1F4.AA. 2F1.C56. 2FD.1E2. 1FF7374. 1F652. 1F4. 1F4. 9F92.  \$z \$172  \$2 \$172
38B 86 86 86 86 86 86 86 86 86 86 86 86 86	34 \$167  34 \$167  34 \$167  35 \$167  36 \$167  36 \$167  36 \$167  37 \$167  38 \$167	\$233871 ED22CC FF. 4F4 F7. F6. E6. E6. E6. E6. E7. 11282 .5542. 4541 \$2. 4541 \$2. 4541 \$2. 4541 \$2. 4541 \$3. 4170	\$\frac{14971}{\tag{1}}\$ \$\tag{44} \tag{28} \tag{45}\$ \$\tag{84} \tag{46}\$ \$\tag{485} \tag{384}\$ \$\tag{485} \tag{385} \tag{3865}\$ \$\tag{885} \tag{385}\$	7822882 . 7822882 . 1FB11CD1 . 1F9 2F6 . 1F4 AA . 2F7 C6 . 2FD 1E2 . 1F7374 . 1F652 . 1F4 . 9F92 \$z \$172 \$z \$172 \$z \$172
\$\frac{3BB}{8C}\$ \$\frac{8C}{8C}\$ \$\frac{8C}{8C	32163.361 .2F337F6728D1 .FD. BF2.2F7 .FT. TFC9 .F5.5FB9 .F5.5FB9 .F5.5FB9. .68733897.6A75. 32 #167 34 #167 35 #167 36 #167 36 #167 37 #167 38 #167	\$x \$170  \$x \$170  \$x \$170  \$x \$170  \$x \$170  \$x \$170  \$x \$202  \$3542  \$4547  \$656  \$666  \$	\$y \$171  \$y \$171  \$4, 28E1  .5B2E8  .8AAC  .6B8BA9  .7E21D3  .CD126A  .352  .352  .352  .352  .352  .253  .261  .361  .375  .384  .4AB538A4  .5D884  .73C  .9988  .8E93  .8188457  .3382  .28666	7822882 .7822882 .1FB11CD1 .1F9, 2F6. .1F4. AA. .2F1. C6. .2FD. 1E2. .1F7374 .1F652 .1F4. .9F92. \$z #172
38B 86 86 86 86 86 86 86 86 86 86 86 86 86	34 \$167  34 \$167  34 \$167  35 \$167  36 \$167  36 \$167  36 \$167  37 \$167  38 \$167	\$233871 ED22CC FF. 4F4 F7. F6. E6. E6. E6. E6. E7. 11282 .5542. 4541 \$2. 4541 \$2. 4541 \$2. 4541 \$2. 4541 \$3. 4170	\$\frac{14971}{\tag{1}}\$ \$\tag{44} \tag{28} \tag{45}\$ \$\tag{84} \tag{46}\$ \$\tag{485} \tag{384}\$ \$\tag{485} \tag{385} \tag{3865}\$ \$\tag{885} \tag{385}\$	7822882 . 7822882 . 1FB11CD1 . 1F9 2F6 . 1F4 AA . 2F7 C6 . 2FD 1E2 . 1F7374 . 1F652 . 1F4 . 9F92 \$z \$172 \$z \$172 \$z \$172