A Feeble Classifier Relying on Strong Context

A. Lawrence Spitz

Document Recognition Technologies, Inc.
616 Ramona Street, Suite 20
Palo Alto, California 94301, USA
E-mail: spitz@docrec.com

1 Introduction

Henry Baird has described the application of character shape coding to particular tasks as using a feeble classifier in a strong context. While perhaps not appropriately respectful of the profound innovation demonstrated by this technology development, he is not wrong.

Character shape coding is a technology that was developed specifically to show robust performance in the presence of poor image quality and has been shown to perform quite well in that domain. Even a simple classifier can perform adequate recognition to serve many applications if enough contextual information of the right sort is given.

Character shape coding can be used to serve a broad range of applications from language identification where the patterns of occurrence of different character shapes can readily be used to identify which of dozens of languages are present, to postal address reading where the tightly constrained lexicon reduces ambiguity by large factors to the attempted reconstruction of document textual content.

However, shape coding is not perfect and we will examine its weaknesses in an attempt to define a text image recognition environment where algorithmic failure, while not guaranteed, is highly likely.

Perhaps if we can fool a feeble classifier we can also fool more sophisticated ones. Even a feeble classifier can perform adequate recognition to serve many applications if enough contextual information is given. To support a Human Interactive Proof application we must either degrade the source of information of provide context in such a form that algorithmic pattern recognition can not be accomplished in a constrained time.

2 Character shape coding

It can be shown that such a simple classifier actually has some advantages over more complex ones, not only in terms of reduced computational cost, but in robustness in the presence of noise.

The process of generating Character Shape Codes (CSCs) encodes only the grossest features of character image size and placement, while ignoring the high spatial frequency features that Optical Character Recognition (OCR) relies upon. Because character shape coding is a lossy transform resulting in an ambiguous representation of the characters in the document, it is important that robustness be maintained at as high a level as possible lest the representation become meaningless. The robustness required varies with the particular application, but whether that application is word spotting, language identification, information retrieval or support for word recognition, tolerance for a range of production values and types of noise present in document images is quite important.
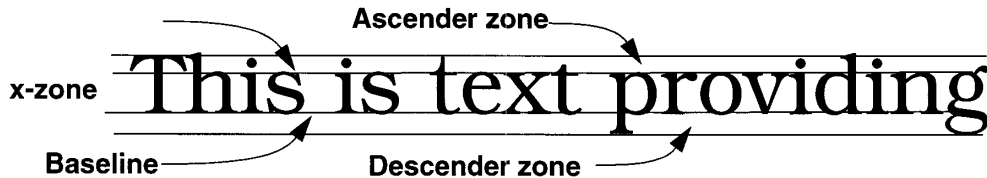
In its simplest implementation, CSCs are assigned as follows:

CSC	Characters	Definition
A	A-Zbdfhklt	Character extends into ascender-zone
x	acemnorsuvxwz	Character contained in x-zone
g	gpqy	character occupiex x-zone plus descender-zone
i	i	x character with a single mark above it
j	j	descender character with a mark above

An extension to the process of Character Shape Coding is the aggregation of CSCs bounded by white space or punctuation into Word Shape Tokens (WSTs).

3 Applications

Character shape coding has been applied to several different applications including duplicate document detection, word spotting, information retrieval, document content characterization and document topic identification. Three other applications will be briefly described.

3.1 Language Identification

It is possible to identify languages set in Roman type either on the basis of frequently occurring WSTs or on the distribution of CSC n-grams.

Token	English			French			German		
	Rank	Occ	Word(s)	Rank	Occ	Word(s)	Rank	Occ	Word(s)
AAx	1	8.1	the, The						
ix	2	4.1	is, in				4	3.0	im, in
Ax	3	3.8	to	1	14.4	la, le, du			
xA	4	3.5	of						
xxA	5	2.9	and				3	3.3	auf
Axx				2	7.7	les, des	1	8.6	der, das
xx				3	3.7	en			
Aix							2	5.3	die, Die

Confusion in language identification tends to reflect language similarity such as Spanish/Portuguese, the Nordic or Slavic languages. It is interesting to note that the expected confusion between Dutch and Afrikaans is not found due to the frequently occurring word 'n in Afrikaans which has a distinctive word shape.

Language of Document \ Detected Language	en	ge	du	af	fr	it	sp	pt	ru	da	no	se	ic	ga	we	cr	cs	pl	hu	fi	tu	sa	vi	Error Total
en	36	1			1																			2
ge		29												1										1
du			28																					0
af			1	29																				1
fr					23			1						1										2
it						35		1	1															2
sp							39	2																2
pt								25																0
ru						3			38															3
da				1						25														1
no											39	2												2
se												40								1				1
ic													23							1				1
ga				2					2				1	31										5
we															30							1		1
cr																33								0
cs								1								6	24	16	3	5				31
pl																		28						0
hu																		2	29					2
fi																	6	2		24				8
tu						1														1	28			2
sa																				1		37		1
vi																							13	0
Error Total	0	1	1	0	2	5	1	4	4	0	0	0	3	2	0	6	6	18	3	9	1	1	0	69

3.2 Postal Address Reading

In an application such as postal address reading there is a rigidly constrained context. The list of legitimate city and street names is relatively small. For example, there are 15130 distinct city names in Germany, 198 of which have the most frequent WST: **AxxAxx.** Performing OCR only on selective character positions quickly further reduces this list to a single city name. If the WST is **AAAxxAxxxxx**, then there are only four related city names and if the WST were **AAxAAgxxA**, then the unique city name, Stuttgart, is obtained without any OCR.

Application of shape coding to German postal addresses resulted in a speed up of 30 times in city name recognition and 130 times in street name recognition.

```
AxxAxx (198):

Aachen Aschau Auufer Bachra Bandau Bandow Barkow Bartow Bauler Beckum Beelen
Benken Berkau Beuden Bochow Bochum Bockau Borken Borkow Borkum Borlas Borler
Bredow Brehme Brehna Brodau Buchen Buckau Buckow Burkau Buskow Carlow Cochem
Daaden Dachau Damlos Daskow Dechow Demker Derben Deuben Dorfen Drehna Duckow
Eschau Freden Frehne Gamlen Gartow Gerdau Geslau Gnadau Gorden Goslar Grabau
Graben Grabow Gruhno Gumtow Gustow Hartau Heeßen Herten Horben Hosten Jeeben
Jeetze Jucken Kantow Karben Karlum Kemtau Kerben Kerkau Kerken Kerkow Kesten
Konken Krahne Krakow Krebes Krokau Krukow Kuchen Laaber Lachen Landau Lankau
Laskau Lastau Lauben Laufen Lauter Leetza Leuben Lochau Lochum Lostau Losten
Luchau Luckau Luckow Lunden Macken Manker Markee Marlow Mauden Mechau Mechow
Meeder Menden Merkur Mochau Mochow Muchow Neetze Neußen Neufra Neuler Norden
Norken Ornbau Panker Panten Parkow Parlow Pastow Pechau Penkow Penkun Perkam
Perlas Pockau Pomßen Postau Pratau Preten Profen Puchow Rambow Randow Rantum
Rastow Reeßum Reußen Reuden Rochau Rockau Roskow Ruchow Saadow Saalau Saalow
Sachau Sandau Santow Saxler Seelen Seelow Seelze Semlow Senden Sontra Suckow
Tacken Tantow Techau Treben Trebra Trebur Trebus Tuchen Usedom Verden Vreden
Wachau Wachow Wacken Warder Wardow Warlow Wasdow Wauden Wenden Werben Werdau
Werder Werdum Westre Xanten Zachow Zechow Zeetze Zerben Zeuden Zorbau Zuchau

AAAxxAxxxxx (4):
Altenhausen Attenhausen Ettenhausen Effenhausen


AAxAAgxxA (1):
Stuttgart
```

3.3 Reconstruction

Whereas people can quickly glance at a printed document and get the gist of it, computers typically have needed to go through the process of constructing a character-based representation before being able to recognize any of the words on a page. Document Reconstruction is a method for identifying a large fraction of the words present in a document image which sidesteps the need for optical character recognition. Each word shape token in the representation is matched against a lexicon, and a word replaces the word shape token if it matches uniquely, or if it is the most probable candidate based on measures that include word frequency.

Additionally we take advantage of the comparison of character cell bitmaps to identify characters in words with ambiguity. The reconstructed text of the Gettysburg address is shown below.

```
Four score and seven years ago our fathers brought forth upon this
continent a new nation conceived in liberty and dedicated to the
proposition that all men are created equal
    Now we are engaged in a great civil war testing whether that nation or
any nation so conceived and so dedicated can long endure {We,be} are met
on a great battlefield of that war
    {We,be} have come to dedicate a portion of that field as a final
resting place for those who here gave their lives that this nation might
live It is altogether fitting and proper that we should do this
    {But,but} in a larger sense we cannot dedicate we cannot consecrate we
cannot hallow this ground thue brave men living and dead who struggled
here have consecrated it far above our poor power to add or detract thue
world will little note nor long remember what we say here but it can
never forget what they did here
```

3

It is for us the living rather to be dedicated here to the unfinished
work which they who fought here have thus far so nobly advanced It is
rather for us to be here dedicated to the great task remaining before us
that from these honored dead we take increased devotion to that cause for
which they gave the last full measure of devotion that we here highly
resolve that these dead shall not have died in vain that this nation
under God shall have a new birth of freedom and that government of the
people by the people for the people shall not perish from this earth

4 Conclusions

Character shape coding has been shown to have reasonable application to particular applications. It is a feeble classifier but it relies on context to resolve ambiguity. The challenge for HIP is to develop contexts which are unknown or difficult to encode to deny the use of contextual information to algorithmic exploitation.