Copyright Protection
for the
Electronic Distribution of Text Documents

J. T. Brassil
Bell Labs, Lucent Tech.
Murray Hill, N.J.

S. Low
Univ. of Melbourne
Parkville, Vic., Australia

N. F. Maxemchuk
AT&T Labs. - Research
Florham Park, N.J.

ABSTRACT

Each copy of a text document can be made to be different in a nearly invisible way by repositioning or modifying the appearance of different elements of text: lines, words, or characters. A unique copy can be registered with its recipient, so that subsequent unauthorized copies that are retrieved can be traced back to the original owner.

In this paper we describe and compare several mechanisms for marking documents and several other mechanisms for decoding the marks after documents have been subjected to common types of distortion. The marks are intended to protect documents of limited value, that are owned by individuals who would rather possess a legal than an illegal copy if they can be distinguished. We will describe attacks that remove the marks and countermeasures to those attacks.

An architecture is described for distributing a large numbers of copies without burdening the publisher with creating and transmitting the unique documents. The architecture also allows the publisher to determine the identity of a recipient who has illegally redistributed the document, without compromising the privacy of individuals who are not operating illegally.

Two experimental systems are described. One was used to distribute an issue of the IEEE Journal on Selected Areas in Communications and the second was used to mark copies of company private memoranda.

1. Introduction

Copyright protection is becoming more elusive as computer networks such as the global Internet are increasingly used to deliver electronic documents. Document distribution by network offers the promise of reaching vast numbers of recipients. It also allows information to be tailored and preprocessed to meet the needs of each recipient. However, these same distribution networks represent an enormous business threat to information providers — the unauthorized redistribution of copyrighted materials.

Information providers seek to use computer networks such as the global Internet as a low-cost distribution medium for documents which they sell in other forms (e.g. magazines, newsletters, books, CD-ROMs). But since copies of digital documents are indistinguishable from the original, and wide redistribution on networks is easily accomplished, the threat of information piracy is real. This threat must be reduced in order for networks to achieve their potential.

Rather than attempt to *prevent* unauthorized document copying and dissemination, we propose technology to discourage it. Our emphasis is strictly on "commercial grade" document security; we focus on security techniques which are simple to implement, rather than those that are extremely resistant to attack. Our security goals are modest; we hope to make unauthorized copying and dissemination of electronic publications at least as difficult as if the publications were distributed on durable media (e.g. paper, disc). If publishers can easily use computer networks for document distribution, with no additional fear of revenue loss due to "bootlegging" than they already face, then we have achieved our goal.

In this paper we discuss ways to discourage redistribution of one important class of documents, namely formatted, black and white text images. Each document recipient (i.e. subscriber) receives a document containing a unique set of *marks*[1]. Each mark corresponds to an imperceptible manipulation of the text. We will show that information hidden in this fashion can be reliably recovered, even from severely degraded copies.

Adding a unique marking to a document can serve many purposes. For applications such as copyright protection, the marks placed in an image can be used as an "identifier" of the rightful recipient of a document. Through analysis, any recovered marked document can be associated or "traced" to the original, intended recipient. In addition, personal information, which a document recipient may be unwilling to make public (e.g. a telephone or credit card number), can also be included in an image. Indeed, being aware that such information can be hidden makes a document more "valuable" to a recipient. The original recipient is unlikely to give his credit card number to someone who receives illegal copies of documents. In addition to discouraging copying of electronic documents, hidden identifiers can also used to deter "leaks" of paper copies of closely held executive correspondence. If an illegal copy of a private report is found, the original recipient can be identified.

The technology that converts paper documents to electronic documents is increasing in use and decreasing in cost. Many companies routinely scan paper documents into their computer systems where they can be stored, filed and retrieved more easily. The cost of facsimile machines, that convert from paper to electronic form, has decreased and are used in many homes. Similar advances have been made in computer printers. In fifteen years printers have evolved from expensive phototypesetters that required special paper and resided in computer centers to devices on home computers that can print in color on almost anything. Eventually paper documents will be scanned and distributed over communications networks almost as easily as electronic documents. At this point, the threat of information piracy will be similar for electronic and paper documents. The techniques that we are developing for electronic documents will also be needed to protect conventional printed material, such as newspapers or magazines.

This paper explores the technology, systems, applications, and issues associated with hiding information in document images. The remainder of the paper is organized as follows. In the next section we introduce a collection of techniques for embedding indiscernible marks in formatted documents. We also discuss techniques for detecting the presence of these marks in documents, including documents which have been degraded by "noisy" document reproduction devices such as copiers and facsimile devices. Marks that are encoded into the appearance of text may always be removed by retyping the text. As a partial countermeasure against this attack we make it unlikely that a mark can be replaced by another valid mark, so that we can detect when documents have been modified. Section 3 describes the architecture and design of a network distribution system which addresses the scaling issue in distributing uniquely marked documents to a large number of recipients. When publishers register documents with recipients, privacy becomes an issue. In section 3 we discuss techniques to protect both the rights of the publishers and the privacy of the individual. Two experimental document marking systems that we have implemented are described in Section 4.

2. Text Marking and Detection Techniques

A mark can be placed in a *formatted* document by altering either the appearance or the position of a text element such as an individual character or word. We say that a collection of marks within a document forms a *codeword*, and that the document is encoded. In this section we introduce three distinct types of marks. Electronic copying does not alter the marks, but conventional copying techniques degrade the image and may make the marks unreadable. The challenge is to find imperceptible text alterations that can be reliably detected after documents have been printed, photocopied, or transmitted by facsimile.

2.1 Marking Techniques

The three marking techniques that we propose illustrate different approaches rather than form an exhaustive list of marking techniques. They can be used either separately or jointly. Each technique enjoys certain advantages or applicability as we discuss below.

2.1.1 Line-Shift Coding In this approach a mark is embedded on a page by vertically displacing an entire text line, as in figure 1. In a typical implementation, a line is moved up or down, while the line immediately above or below (or both) are left unmoved. These unmoved adjacent lines serve as reference locations in the decoding process.

the Internet aggregates traffic flows from many end systems. Understanding
effects of the packet train phenomena on router and IP switch behavior
will be essential to optimizing end-to-end efficiency. A range of interesting

the Internet aggregates traffic flows from many end systems. Understanding
effects of the packet train phenomena on router and IP switch behavior
will be essential to optimizing end-to-end efficiency. A range of interesting

Figure 1. Example of line-shift coding. In the first group of 3 lines, the middle line has been shifted down by 1/300 inch. To demonstrate this displacement the 3 lines are repeated with the middle line printed both in the unshifted and shifted positions, resulting in its apparent emboldening.

Most documents are formatted with uniform spacing (i.e. "leading") between adjacent lines within a paragraph. Though the human eye is particularly adept at noticing deviations from uniformity, our experience suggests that vertical line displacements of 1/300 inch and less go unnoticed by readers.

The principal advantage of this marking technique is found in decoding. Since a document's initial interline spacing is uniform, the presence or absence of a mark can be detected by analysis of the interline spacing of a recovered document, with no need for any additional information about the original, unmarked document. Therefore, anyone can read the information. This technique can also be used to include computer readable information in books or journals for cataloging and automatic identification.

2.1.2 Word-Shift Coding In this technique a mark is embedded by horizontally shifting the location of a word within a text line, as shown in figure 2. In a typical implementation, a word is displaced left or right, while the words immediately adjacent are left unmoved. These unmoved words can then serve as reference locations in the decoding process.

the Internet aggregates different sessions from many end systems. Understanding
the Internet aggregates different sessions from many end systems. Understanding
the **Internet** aggregates **different** sessions **from** many **end** systems. Understanding

Figure 2. Illustration of word-shift encoding. The first line is unshifted; the second line contains 4 words each shifted by 1/150 inch. Note that the word spacing in both lines appears natural. The third line is an overlay of the first two lines.

Formatted documents with justified text typically use variable spacing between words to distribute white space in a visually pleasing fashion. Readers accept a wide variation in text setting within a line; our experience suggests that horizontal word displacements of 1/150 inch and less readily go unnoticed. Since the word spacing in the original document is not uniform, detecting a word displacement requires knowledge of the original word spacing. Hence, the word positions in the unmarked document must be known in order to extract the hidden information. The hidden information can only be read by the organization that owns the original document or its agent.

2.1.3 Character Coding Character coding is a class of techniques which embed a mark by altering a particular feature of an individual character, as shown in figure 3. Examples of possible feature alterations include a change to an individual character's height, or its position relative to other characters (e.g. a "kerning" adjustment). Once again, some character features are left unchanged to facilitate decoding. For example, a detection algorithm might compare the height of a hypothetically altered character with that of another unmodified instance of the same character elsewhere on the page.

Figure 3. Example of character coding. In the first line, the letter '**r**' in the word 'Internet' had been shifted down by 1/600 inch. The second line reveals the displacement by reprinting the word with a larger font size.

Imperceptibly embedding a mark by character alteration often requires extremely careful attention to the context of the character to be altered. A reader is more likely to notice a character alteration if an identical, unaltered character is immediately adjacent. Detecting the presence or absence of a mark might or might not require information from the original, unmarked image, depending on the marking technique and the rule for selecting the characters that are altered.

2.1.4 Comparison The marking techniques introduced above alter ever smaller textual elements to embed a mark; the size of the alterations is similar but the "signal level" is greater for the large elements. Since the larger text elements have a larger signal to noise ratio when subjected to the same distortions, we expect line-shifting to be the most robust marking technique. For this reason, line-shifting is particularly well suited to marking documents to be distributed in paper form, where considerable degradation can be introduced by photocopying, or simply handling.

We also expect that word-shifting will be less discernible to the reader than line-shifting, yet also more difficult to detect in the presence of noise. Character encoding has the advantage of a potentially large coding *density*; on a given page of text, many more marks can be inserted by altering characters than by altering lines or words. This property makes this technique attractive in applications requiring very wide distribution of uniquely marked electronic documents (e.g. distributing a large circulation, general interest magazine). The large coding density also allows redundancy to be added for error correction.

2.2 Document Marking System

A document marking system comprises an encoder and a decoder, which we describe in the next two subsections. The encoder uses one or more of the marking techniques introduced in the previous section to embed a codeword in a document. The decoder analyzes a recovered (and possibly degraded) document image and extracts the embedded codeword. Henceforth we will limit our discussion to line-shift and word-shift encoding.

A document can be represented in many forms, such as an "image" (i.e. bitmap) or in a page description language such as PostScript[1]. Marks can be embedded in a document taking any form. For the following discussion, we will assume that we are marking a Postscript document.

1. PostScript is a trademark of Adobe Systems, Inc.

2.2.1 Encoder An encoder comprises a document preprocessor and a word shifter (Figure 4). The preprocessor is an off line, one-time operation which performs two services:

1. The preprocessor replaces arbitrary PostScript commands with a set of equivalent Postscript commands which are easily recognized and manipulated by the word shifter.
2. The preprocessor identifies which text words can be shifted, ensuring good decoding performance with little perceptual change to the document.

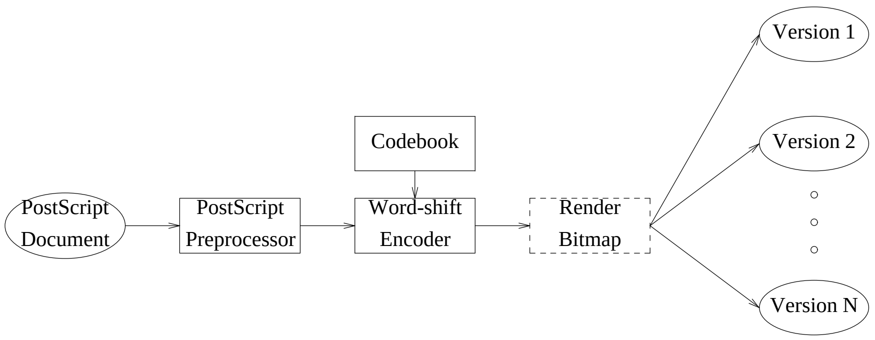


Figure 4. The logical architecture of an encoder.

The first step is necessary because different word processors generate different sequences of postscript commands to place characters in the same position on a page. Following preprocessing, each text word which is a candidate for displacement is set at an x,y location on a "virtual page". The word shifter modifies the x-coordinate of a word (for word-shift encoding) or the y-coordinate of a text line (for line-shift encoding). The correspondence between the word or line displacements and codewords is maintained in the codebook.

2.2.2 Decoder The decoding process is depicted in Figure 5. There we assume that a single page of a degraded paper copy of an encoded document is recovered. The page is scanned at an 8-bit depth (i.e. grayscale) and standard image processing operations are applied to the image primarily to reduce noise and prepare the image for further processing. These operations typically include edge cropping, binarization, salt-and-pepper noise reduction, deskewing (rotation of text lines to the horizontal), and thinning (of individual characters). The locations of text elements are extracted from the resulting image. These locations are then compared with the locations of text within each of the originally distributed, unique documents. A decoder or decision algorithm finds that the recovered image matches one of the original documents most closely, and refers to a codebook to find the identity of the original document recipient.

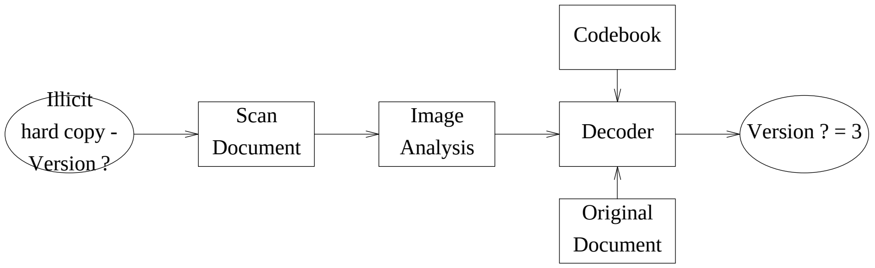
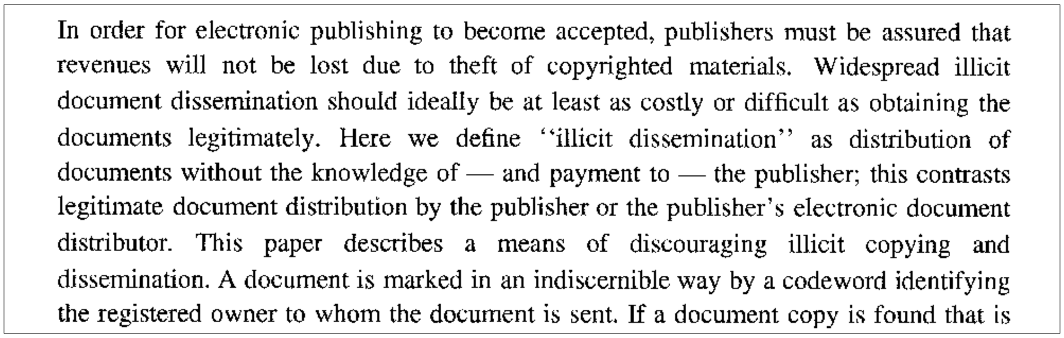


Figure 5. The logical architecture of a decoder.

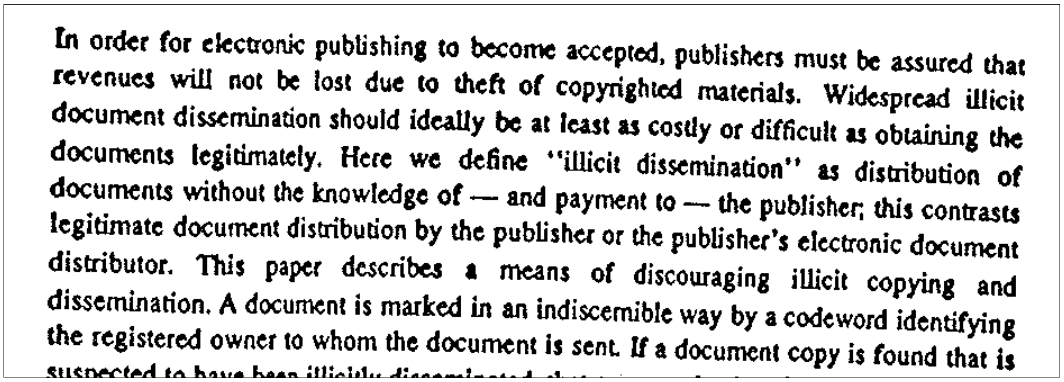
2.3 Image Analysis

The first step in the decoding process is to remove the noise and distortion that we expect to find in a recovered image. We begin with a brief discussion of this noise and its sources.

2.3.1 Noise Document images are often reproduced by devices (e.g. facsimile, plain paper copier) that are effectively modeled as noisy communication channels. A sample of an original document and its tenth copy is shown in figure 6. In this section we briefly examine defects these devices impart on document images, and how to create robust watermarks which can be detected in the presence of these defects.



In order for electronic publishing to become accepted, publishers must be assured that revenues will not be lost due to theft of copyrighted materials. Widespread illicit document dissemination should ideally be at least as costly or difficult as obtaining the documents legitimately. Here we define ''illicit dissemination'' as distribution of documents without the knowledge of — and payment to — the publisher; this contrasts legitimate document distribution by the publisher or the publisher's electronic document distributor. This paper describes a means of discouraging illicit copying and dissemination. A document is marked in an indiscernible way by a codeword identifying the registered owner to whom the document is sent. If a document copy is found that is



In order for electronic publishing to become accepted, publishers must be assured that revenues will not be lost due to theft of copyrighted materials. Widespread illicit document dissemination should ideally be at least as costly or difficult as obtaining the documents legitimately. Here we define ''illicit dissemination'' as distribution of documents without the knowledge of — and payment to — the publisher; this contrasts legitimate document distribution by the publisher or the publisher's electronic document distributor. This paper describes a means of discouraging illicit copying and dissemination. A document is marked in an indiscernible way by a codeword identifying the registered owner to whom the document is sent. If a document copy is found that is

Figure 6. Sample of an original text image (upper) and its tenth copy (lower).

Expansion or shrinkage of copy size is present to some degree in nearly every image reproduction device. In some cases size changes are purposely introduced for perceived reproduction quality improvement or as an anti-counterfeiting measure. The expansion along the length and width of a page is typically different.

To counter the effect of page size changes, we encode information differentially and use the relative, rather than absolute, position of textual objects. We can also encode information independently along both the width of the page (i.e. word shifting), and along the length of the page (i.e. vertical shifting of text lines).

Defects also occur on relatively large spatial scales (i.e. $> 1\ cm^2$). One such phenomena observed in recursively copied pages is "baseline waviness" (i.e., text rising above and/or falling below the logical line on which text sits). To counter such large-scale spatial defects, our differential encoding mechanisms use textual objects which are relatively close (e.g. adjacent text words, or characters in close proximity). In addition, a large number of text objects are left untouched to serve as reference points and to provide an estimate of distortion to improve the detection.

Many other image defects are readily observed but do not appear to have a dramatic effect on our detection results[2]. This includes salt-and-pepper noise, some of which is easily removed by standard picture processing techniques. Linear text line skew (i.e. a tilted line) is approximately corrected by image rotation. Both edge raggedness (i.e. blurring) and fading have surprisingly little consequence in detection performance. Other researchers have concluded that blurring tends to be isotropic about each text character[3]. Because blurring is isotropic, it does not significantly alter the position of the center of mass or *centroid* of a text object, which we use for decoding.

Image distortion is usually more severe in one dimension — either along the length or the width of a page — than the other. This is typically the "paper direction," or the orientation of paper moving though an image reproduction device. Variable paper thickness, drums and wheels out-of-round, etc., all contribute to nonconstant paper speed that results in more distortion in the paper direction. Note that a paper direction along the width of a page will have more of an adverse affect for word shift encoding while a paper direction along the length of a page will have more of an adverse affect for line shift encoding. Since a recovered document may have been reproduced on different devices with different paper directions, a marking system should encode the same information along both directions to increase decoding performance.

2.3.2 Profiles The second step in the decoding is the conversion of the noise-reduced image to a form from which text locations can be easily identified. All the detection methods to be described detect marks by creating and analyzing a *projection profile* of a page image. We now explain what a profile is and discuss how the noise in an image affects a profile.

Following digitization a page image is represented by a 2-dimensional array with elements

$$f(x,y) \quad x = 0, 1, \ldots, W, \quad y = 0, 1, \ldots, L \ ,$$

where $f(x,y)$ represents the intensity of the pixel at position (x,y). For a black and white image, $f(x,y) \in \{0,1\}$. Here, W and L, whose values depend on the scanning resolution, are the width and length of the image in pixels, respectively. Each array row corresponds to a horizontal scan line in the scanned page image. A subimage containing a single text line is simply the subarray

$$f(x,y) \quad x = 0, 1, \cdots, W, \quad y = t, t+1, \cdots, b,$$

where t and b are rows in the image above and below the text line. For instance, we may take t or b to be at the mid-point of the interline spacing above and below the line, respectively.

A profile is the projection of a 2-dimensional array onto a single dimension. The *horizontal profile* of the subarray containing the text line is

$$h(y) = \sum_{x=0}^{W} f(x,y), \quad y = t, t+1, \cdots, b \ ,$$

i.e. the sum of array elements along each row y. The *vertical profile* of the subarray is

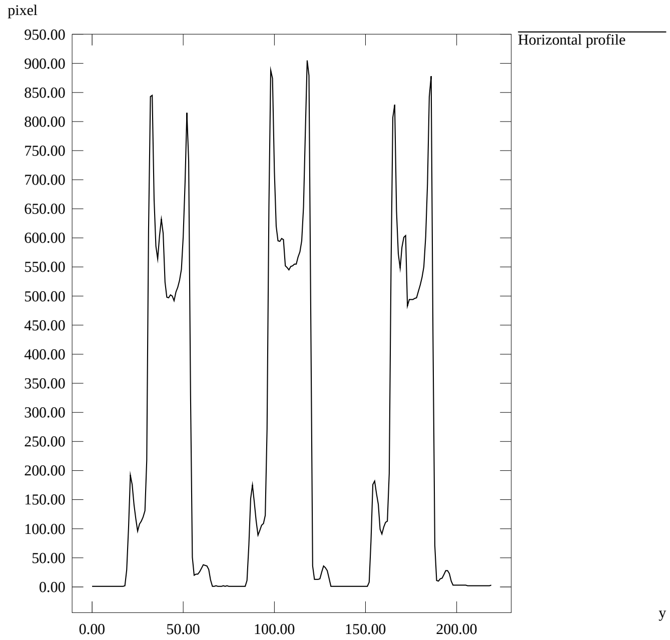
$$v(x) = \sum_{y=t}^{b} f(x,y), \quad x = 0, 1, \cdots, W \ ,$$

i.e. the sum of array elements along each column x.

Figure 7 shows a horizontal profile of three text lines, and a vertical profile of a single text line containing six words. Note the different scales on the two profiles. A horizontal profile consists of distinct ‘columns’ and ‘valleys’. The ‘columns’ correspond to horizontal scan lines through a text line, and the ‘valleys’ to interline spaces. The width of each column in the horizontal profile corresponds to the "body height" of characters on a text line; this is about 41 pixels at a 10 point font size (i.e. 10/72 inch character height) scanned at 300 pixels/inch. The vertical profile of a single text line has shorter columns and narrower valleys that are much less pronounced. The height of a column corresponds to a vertical scan line through a single character, which is no larger than the font’s body height in pixels.

A profile is an integer-valued vector, though our figures often depict it as a continuous function, and we often find it mathematically convenient to approximate it as one. A profile contains information about the relative locations of text in an image, but is far more convenient to work with than the image.

Following the removal of noise from a recovered image, a profile is compiled. We assume that any remaining noise or distortion that exists in the image, such as translation and scaling, affects neighboring regions of the image in a similar fashion. We refer to rectangular image regions as blocks. We say a block



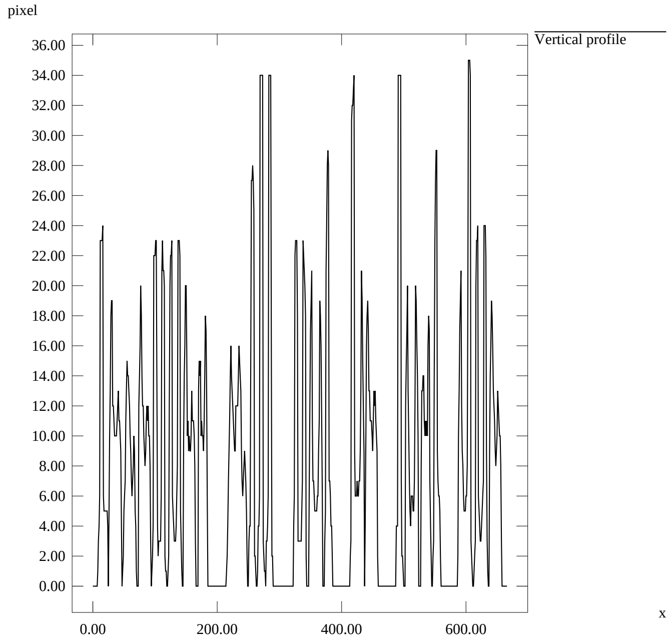


Figure 7. Horizontal (upper) and vertical (lower) profile (resolution = 300 dots-per-inch).

is marked if it contains repositioned text which serves as a mark; an unmodified block which serves as a reference is a control block. Control blocks can be used to estimate structural distortions and the correlation structure of the remaining noise. These estimates can be used to remove noise and distortion from marked blocks.

Developing a useful analytical model of each of the many types of noise that affect a document image would be challenging. For simplicity we assume that after we remove the correlated distortion that is caused by the processes that we have described, a profile $h(y)$ on some interval $[b,e]$ is corrupted only by additive noise $N(y)$, i.e.

$$g(y) = h(y) + N(y), \quad y = b, \cdots, e,$$

where $N(y)$ are independent and identically distributed (i.i.d.) zero-mean Gaussian random variables. This white Gaussian noise models all the distortions not accounted for as well as errors introduced by any compensation we make. A sample of noise $N(y)$ measured from a horizontal and a vertical profiles is shown in Figure 8. The corresponding empirical distributions of $N(y)$ is shown in Figure 9. These figures suggest that the Gaussian model is a reasonable first approximation.

2.4 Detection

The third step in the document decoding process is the decision (detection) step. Here an algorithm takes as input the profile of a recovered, marked image, and in some cases the profile of the original, unmarked image. The output of the detection step is our hypothesis of the codeword most likely embedded in the recovered image.

We begin with some notation. Let $h(y)$ be the profile of an original document page (i.e. unmarked by any text displacement). Consider the three profile intervals $[b_1, e_1]$, $[b_2, e_2]$, and $[b_3, e_3]$; where b_i is the beginning of the block and e_i is the end of the block. In line-shift encoding, each block encompasses a text line; in word-shift encoding, each block encompasses a group of words. The middle block, $[b_2, e_2]$, is the marked block and the other two adjacent blocks are the control blocks. We assume that the profile height $h(y) = 0$ between the blocks. We mark the original image by displacing the line or word corresponding to the middle block. When the middle block is left shifted by $\varepsilon > 0$: the resultant (uncorrupted) profile is $h^l(y)$ where,

$$h^l(y) = \begin{cases} h(y), & y < b_2 - \varepsilon \text{ or } y > e_2 \\ h(y+\varepsilon), & b_2 - \varepsilon \le y \le e_2 - \varepsilon \\ 0, & e_2 - \varepsilon < y \le e_2 \end{cases}$$

and when the middle block is right shifted the profile is $h^r(y)$ where

$$h^r(y) = \begin{cases} h(y), & y < b_2 \text{ or } y > e_2 + \varepsilon \\ 0, & b_2 \le y < b_2 + \varepsilon \\ h(y-\varepsilon), & b_2 + \varepsilon \le y \le e_2 + \varepsilon \end{cases}$$

An imperceptible text displacement corresponds to a value of the shift ε much smaller than the interblock spacing. The profile $g(y)$ compiled from the noisy, recovered copy (after distortion compensation) is:

$$g(y) = h^l(y) + N(y), \quad y = b_1, \cdots, e_3$$

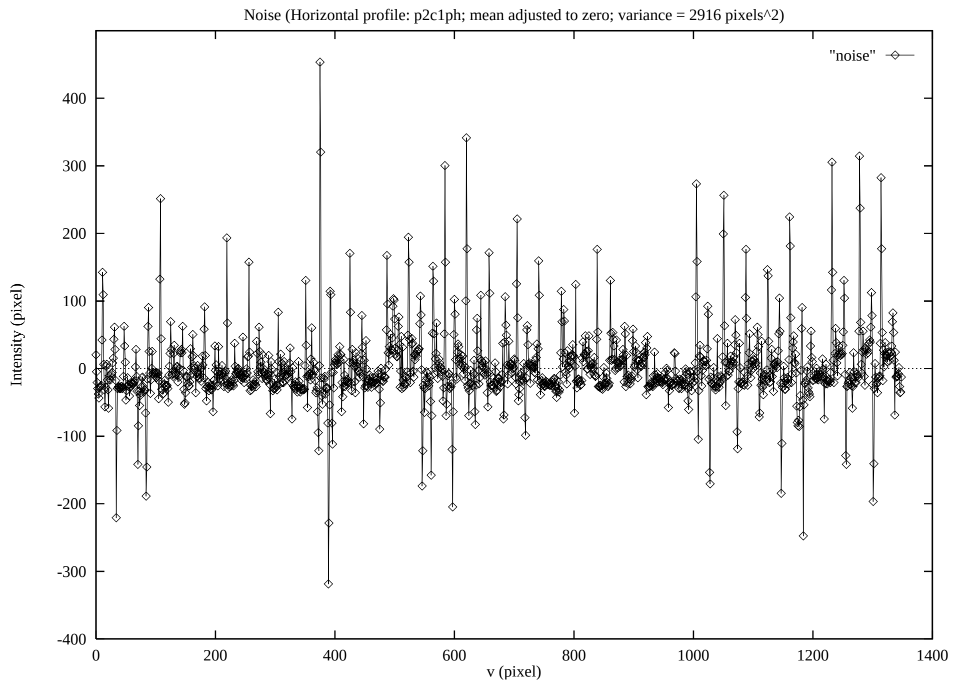
if the middle block is left shifted, and

$$g(y) = h^r(y) + N(y), \quad y = b_1, \cdots, e_3$$

if it is right shifted. We assume that the noise $N(y)$ is Gaussian with zero mean and variance σ^2.

Our task is to decide whether the middle block is left or right shifted given the marked and corrupted profile $g(y)$. Different detection methods use the profiles in different ways.

2.4.1 Feature detection Feature or edge detection attempts to identify the position of a ‘column’ in a profile by estimating the position of a feature in that column. A typical feature one might use would be the rising or falling edge of a column. But by examining the upper profile of Figure 7, one observes that while the column edges are not well-defined, there are two prominent peaks. These peaks correspond to scan lines



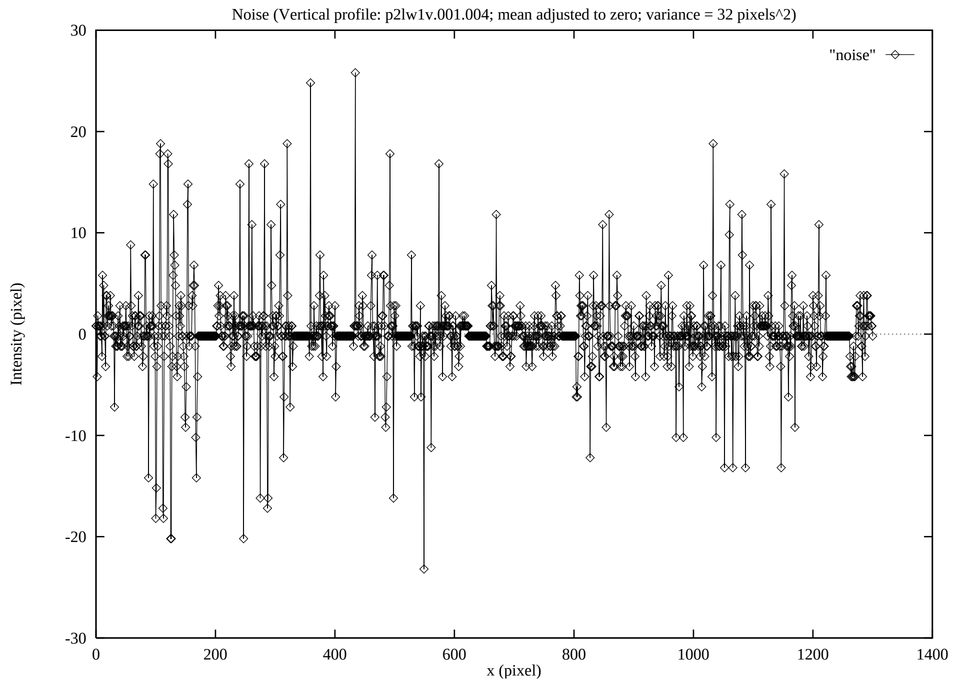
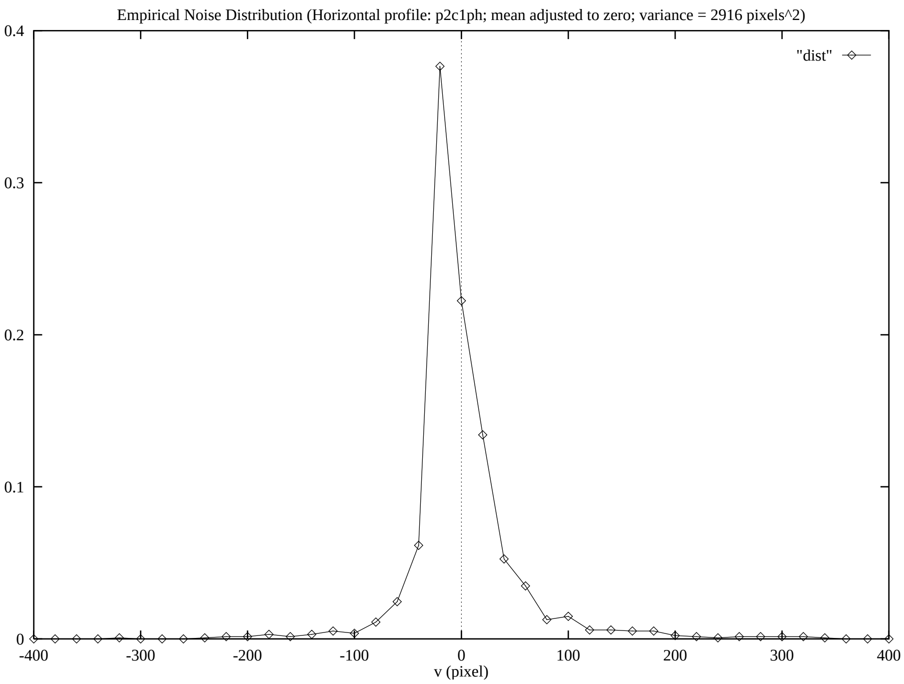


Figure 8. Sample profile noise measured from a horizontal (upper) and a vertical (lower) profile.

through the midline and baseline of the text line; the left peak is caused by horizontal "bars" across the middle of characters such as **A** and **e**, while the right peak is the result of character "feet" in a serif font.

We have developed feature detectors which attempt to locate the position of the text baseline in line-shifted documents (i.e. baseline detection). In documents with initially uniform interline spacing, a baseline detector makes a decision entirely based on the observed profile $g(y)$, without need for the original profile



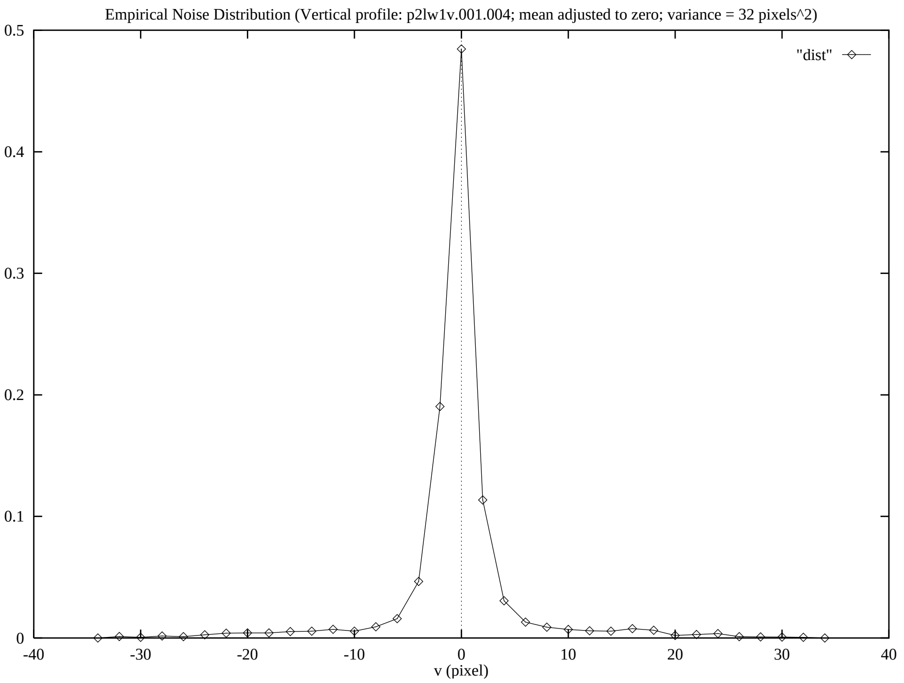


Figure 9. Corresponding empirical distributions.

$h(y)$. In terms of the above model, the baseline l_i of block i is where the profile peaks in a neighborhood $N(e_i)$ of e_i, the right endpoint of the interval $[b_i, e_i]$:

$$l_i = \arg \max_{y \in N(e_i)} g(y), \quad i = 1,2,3.$$

The decision rule a baseline detector implements is simply

Baseline Detector: *Given the observed profile $g(y)$:*

decide line 2 shifted left *if* $l_2 - l_1 < l_3 - l_2$
decide line 2 shifted right *otherwise*

It is also possible to write a feature-based decision rule for the case where the original text spacing is not uniform, as is typical for the spacing between words. Such a detection rule requires the original document profile, or more precisely, the location of features within the original profile.

Feature detection requires the presence of an easily detectable feature. The absence of such a feature in a profile leads us to study other detection algorithms.

2.4.2 Correlation detection A well known result from communication theory tells us that a correlation detector (as might be implemented by a matched filter) optimally detects signals in the presence of additive white, Gaussian noise. Here we take our received signal to be the recovered image profile $g(y)$, and our transmitted signal to be the original profile $h(y)$. A standard derivation (Chapter 4, Reference 4) for a maximum likelihood decision rule yields the following:

Correlation Detector: *Given the observed profile $g(y)$:*

decide line 2 shifted left if $\sum_{b_2}^{e_2} h(y)(g(y-\varepsilon) - g(y+\varepsilon)) \geq 0$
decide line 2 shifted right *otherwise*

Applying this rule is somewhat more involved, since in practice our received signal $g(y)$ suffers from distortions such as expansion and translation. When these effects can be accurately estimated and compensated for, we can reliably detect the direction of shift from the correlation of $h(y)$ and $g(y)$. Note that the decision is based only on the middle block $[b_2, e_2]$, though the probability of detection error depends on the entire profile $h(y)$ and $g(y)$ over all three intervals.[5]

2.4.3 Centroid detection When the effect of translation cannot be compensated for accurately, correlation detection often performs poorly. This motivates the centroid detection method. The observation that a horizontal profile consists of distinct tall and narrow columns suggests the approximation of each column by a delta function situated at the column's centroid; see Figure 7. Marking shifts the centroid of the middle block slightly left or right and leaves the centroids of the control blocks unchanged. The effect of translation of the entire text is eliminated by making detection decision based on the distance of the middle centroid *relative to* its two control centroids.

On the original profile $h(y)$ the centroid of blocks *i* are

$$c_i = \frac{\sum_{b_i}^{e_i} y h(y)}{\sum_{b_i}^{e_i} h(y)}, \quad i = 1,2,3.$$

We can model the displacement of the centroids on the received image profile $g(y)$ by the addition of the random noise component $N(y)$. As a result, the control blocks have centroids

$$U_1 = c_1 + V_1 \quad \text{and} \quad U_3 = c_3 + V_3 .$$

The middle block has been shifted by a size $\varepsilon > 0$ so that its centroid is

$$U_2 = c_2 + V_2 - \varepsilon$$

if it is left shifted, and

$$U_2 = c_2 + V_2 + \varepsilon$$

if it is right shifted. Here, V_i are random variables representing the distortion on the original centroids c_i by the additive profile noise $N(y)$. Since we assume that $N(y)$ is white, the centroid noise variables V_i, $i = 1,2,3$, are independent. Reference 6 shows that the variables V_i are accurately approximated by zero-mean Gaussian with variance ν_i^2 given by

$$\nu_i^2 = \frac{\sigma^2 w_i}{H_i^2} \left[\delta_i^2 + (w_i^2 - 1)/12 \right], \tag{1}$$

where

$$H_i = \sum_{b_i}^{e_i} h(y)$$

$$w_i = e_i - b_i + 1$$

$$\delta_i = c_i - \frac{e_i + b_i}{2}. \tag{2}$$

From (1) the variance ν_i^2 of the centroid noise V_i is not only proportional to the profile noise variance σ^2 but also dependent on the original unmarked profile $h(y)$ through H_i, w_i and δ_i.

To eliminate the effect of translation we base our detection on the distance $U_i - U_{i-1}$ between adjacent centroids instead of the absolute position U_2 of the middle centroid. It is convenient to use as decision variable the differences

$$\Gamma_l := (U_2 - U_1) - (c_2 - c_1)$$
$$\Gamma_r := (U_3 - U_2) - (c_3 - c_2)$$

of the corrupted centroid separations and the uncorrupted separations. Γ_l is the change in the distance of the middle block from the left control block and Γ_r is that from the right control block: without noise $\Gamma_l = -\varepsilon$ and $\Gamma_r = \varepsilon$ if the middle block is left shifted, and $\Gamma_l = \varepsilon$ and $\Gamma_r = -\varepsilon$ if it is right shifted. Then the centroid detection is the following decision rule.

Centroid Detection: *Given the observed values* (γ_l, γ_r) *of* (Γ_l, Γ_r)*:*

decide line 2 *shifted left* if $\gamma_l / \nu_1^2 \le \gamma_r / \nu_3^2$

decide line 2 *shifted right otherwise*

where ν_1^2 and ν_3^2 are the centroid noise variances of the left and right control blocks, respectively, given by (1, 2).

Even though ν_i^2 depends on the profile noise variance σ^2 (see (1)), the detection decision does not, since σ^2 appears in both ν_1^2 and ν_3^2. Only the three parameters H_i, w_i, δ_i of the *uncorrupted* control blocks are necessary for detection. The error probability however does depend on σ^2.

2.4.4 Comparison of detection techniques Feature detection is most directly applicable for detecting line-shifting, since the vertical profile of words do not exhibit a prominent, easy-to-detect feature such as a baseline. The principle advantage of baseline detection is that it operates on just the marked copy and, unlike the other methods, it does not require any information on the original unmarked document. The disadvantage is its relatively poor performance on documents that have suffered significant distortions; see Section III, Reference 7.

Although centroid detection can in principle be applied to detect both line and word spacing, its performance in the presence of noise is satisfactory only for line spacing; see Section V, Reference 6 for experimental evidence and Reference 5 for a theoretical explanation. Compared with baseline detection, it is more reliable but requires centroids of the original unmarked document profile. Compared with correlation detection, it eliminates the effect of translation through differential decoding.

Correlation detection performs much better than centroid detection on word spacing[5]. However its performance is sensitive to how accurately we can compensate for the translation of the profile. This method requires the profile $h(y)$ of the original unmarked document in order to extract the encoded information.

The centroids on the original unmarked document can be treated as a secret (key) that is required for detection. Without the key the marks can be detected using feature (baseline) detection, but only from a clean copy. With the key they can be detected even from a corrupted copy using centroid detection. There are two methods to decode the same encoded information in line shifting. This serves well in a typical scenario where a legitimate document recipient can easily verify the ownership information from the received (clean) copy using baseline detection, while a copyright enforcement agent can reliably extract the identity of the original recipient even from a corrupted illicit copy by using centroid detection.

2.5 Attacks and countermeasures

Marks placed in a text using *any* technique discussed above can be removed by retyping the document, possibly with the help of character recognition devices. By contrast, marks placed in pictures or speech are relatively indelible. More sophisticated attacks attempt to remove the marks without retyping the text nor degrading the quality. If the document is in PostScript format, lines and words are easy to recognize and can be moved to produce random or uniform spacing, while respecting justification. This destroys the marks placed in line and word spacing. If the document is in bitmap format, text lines can be accurately recognized and moved by identifying interline spacing from the document profile. The same attack can also be applied to word spacing, but with less accuracy because of the less prominent valleys and columns in a vertical profile; see Figure 7.

Despite these attacks text marking is well suited for protecting modestly priced documents, such as newspaper or magazine articles. We assume that if authorized and unauthorized copies are distinguishable, and authorized copies are affordable, then most people will not seek out unauthorized copies.

Countermeasures to reduce the threat of tampering include:

- making it more difficult to remove marks,
- making it more expensive to redistribute a bootleg copy than the original,
- making it difficult to forge valid marks.

The distribution mode of a document may make it more or less difficult to remove marks. Marks are relatively easy to remove if the document is distributed in a standard formatting language, such as PostScript, PDF, etc., in which the location of lines and words are explicitly stated and easy to change. It is a bit more difficult to remove marks from a bitmap image, particularly when the text has a number of different font types and point sizes and is mixed in with pictures and figures. As documents become more complicated, automated techniques may require human intervention. Paper copies must first be scanned into a bitmap. The scanned bit maps are noisier than bit maps that have remained in electronic form and are more likely to require human intervention to decode.

One way to make bootlegged copies more expensive is to make it necessary for the bootlegger to transmit more bits than the legitimate publisher. For instance, a publisher may distribute the text and marks in PostScript, but make only the (marked) bitmap version accessible to a user (see below for how this might be achieved). If the bit map has 100 times as many bits as the Postscript version and it takes the publisher 15 minutes to transmit a daily newspaper, then it will take a bootlegger more than a day to transmit the same newspaper.

Finally in many applications the mere ability to recognize illegal copies, rather than the illegal distributor, is a sufficient deterrent. If a document recipient removes or changes the mark, then we cannot determine the original recipient of the document. However, if the mark has not been replaced with another valid mark, then we can determine that the document is not a legally distributed copy. The stigma attached to being caught with an illegal document may be sufficient to discourage people from accepting them.

One method that can be used to make it difficult to forge valid marks is public key cryptography. In order to use a public key, the mark must contain both information specific to the document, such as the title, and the publisher's identification number. The document specific information prevents an attacker from taking a valid mark from one document and placing it on another. The publisher signs the mark with a public key.

In order for signed messages to be secure, they must be long. Otherwise valid messages may be constructed by an exhaustive search. Many marked documents will not be able to encode a large enough number of bits.

A simple method to discourage forging is to randomly select the code words that are used to identify a document from a much larger set. For instance, if we can hide 20 bits in a document, we can construct about 1 million code words. If there are 100 copies of the document, and we select the 100 codewords that are assigned to valid recipients at random, then only 1 out 10,000 codewords is being used. If a recipient changes the codeword on his document, there is only 1 chance in 10,000 that he will pick a codeword that has actually been assigned.

3. Document Distribution

3.1 Push and Pull Systems

Document distribution systems can be classified as push or pull systems. In a pull system the client initiates a request for a transaction. The world wide web is a pull system. In a push system the source initiates the transmission. In the non-electronic domain, a publisher who delivers a newspaper or magazine to the reader's home operates in a push mode. Electronic push systems require a user to be continuously connected to a network, or to be accessible so that a server in the network can initiate a connection. Sending electronic mail to users with workstations that are always turned on and connected to the network is a push system.

Hybrid systems, with both push and pull components are common. A publisher who delivers newspapers to a news stand operates in a push mode, while the reader who visits the news stand to buy the newspaper operates in the pull mode. In the electronic domain, when a computer user receives electronic mail in a mail box that is provided by an ISP, the sender operates in a push mode when the mail is sent and the recipient operates in a pull mode when the mail is retrieved.

Electronic publishing systems should have components of both push and pull systems. A new journal article, or an unexpected news story should be sent to subscribers, while an old article that has been archived should be requested when it is needed. Electronic publishing can operate in a push/pull mode by requiring recipients to check a standard site (news stand) for new issues of a journal. Our initial publishing experiments, such as the JSAC experiment reported in the next section, had to operate in a push/pull mode because the targeted audience was not completely identified, those who were known could not be required to have a continuous presence on the Internet, and, the WEB browser software that comprised the client side software operates in the pull mode.

There are both practical and economic reasons to make the push mode of operation available for electronic publishing. The practical reason is that most people are accustomed to having journals and newspapers delivered to them. They are unlikely to remember to visit a WEB site to retrieve an infrequently published journal and they are unlikely to make time in the morning to download newspaper articles that they are accustomed to reading with breakfast. The economic reasons depend upon the number of users that receive an article. A daily newspaper that is read by several million people at about the same time, on the same day, is delivered more economically in a push mode. The publisher broadcasts the article to all of the recipients rather than transmitting it to each recipient individually as it is requested. The push mode conserves both network bandwidth and the processing performed by the publisher. Journal articles that are archived and are only requested by a few individuals a month use less network bandwidth when they operate in a pull mode and are transmitted only when they are requested.

The current multicast capability of the Internet is adequate for a push system to send a technical journal to several tens of thousands of recipients, world wide once a month. However, a system to deliver a daily newspaper to several million subscribers in a large city every day should use a different technology. For instance, the local cable television network can broadcast the newspaper during the evening so that it is printed at the subscriber’s homes before the next morning.

One of the advantages of electronic publishing is the ability to customize a journal or newspaper to the individual. Broadcasting the entire newspaper does not preclude this possibility, it just changes the location of the filter. In the early proposals, a processor at the publisher filtered all of the news in order to present the articles that correspond with an individual’s profile. In the broadcast system, the news can be filtered by a local, personal computer. Changes in the price of computing makes the latter system a viable alternative.

Changes in the price of computer storage may also affect the way an electronic publishing system is implemented. Clearly, papers that a recipient is not currently interested in reading should not be printed on paper. However, with decreasing disc costs, it may be less expensive for the recipient to archive his journal subscriptions locally rather than paying a publisher to store back issues and deliver individual articles at a later date.

3.2 Marking

In a pull system a publisher can make each document that he sends unique. When a publisher serves an individual client he inserts a mark in the document and registers the document with the recipient. In the JSAC experiment, when a user requested a paper copy of an article we transmitted a postscript file with a unique mark embedded to identify the recipient. The processing needed to generate individual documents was well within the capabilities of our server since line and word positions in postscript are easily modified parameters and we only generated a few thousand papers over the course of a month. Processing at the server does not scale well as the application changes from a monthly technical journal, with a limited readership, to a widely read daily newspaper.

A push system, that uses a broadcast medium to reach a large number of recipients, distributes identical copies of the publication. Marking the copy so that every recipient has a unique copy implies processing the document at the client. Trusted, platform independent interpreters, such as JAVA virtual machine[8], were not widely available when we started the JSAC experiment. JAVA makes it possible to mark the document at the receiver, but also gives the recipient access to the unmarked document.

In reference 9 we describe a system that uses client side software to mark documents without giving the user access to the unmarked document. In figure 10 that system is modified to use multicast and a push, rather than pull, mode. The publisher multicasts the same *encrypted* document to every recipient. The document is in a reasonably compact form, such as LaTex. The publisher unicasts each user a decryption program that contains a unique identification number. The program decrypts the document, inserts the marks corresponding to the recipient’s identification number, and converts the document to a bitmap. If the recipient tries to redistribute the document, not only is it marked, but the recipient must transmit more bits than the publisher.

A recipient can avoid transmitting bitmaps, or even retransmitting the document from the publisher, by giving away the program that decrypts the document instead of the document. An unauthorized recipient can receive the multicast transmission directly from the publisher and use the bootlegged program to construct the document. This tactic can be discouraged by requiring personal information about the original recipient in order to use the program. For instance, the secret key in the decryption program may be included in a function with the recipient’s credit card number. Once again, it is unlikely that a legal recipient will give his credit card number to anyone who steals documents.

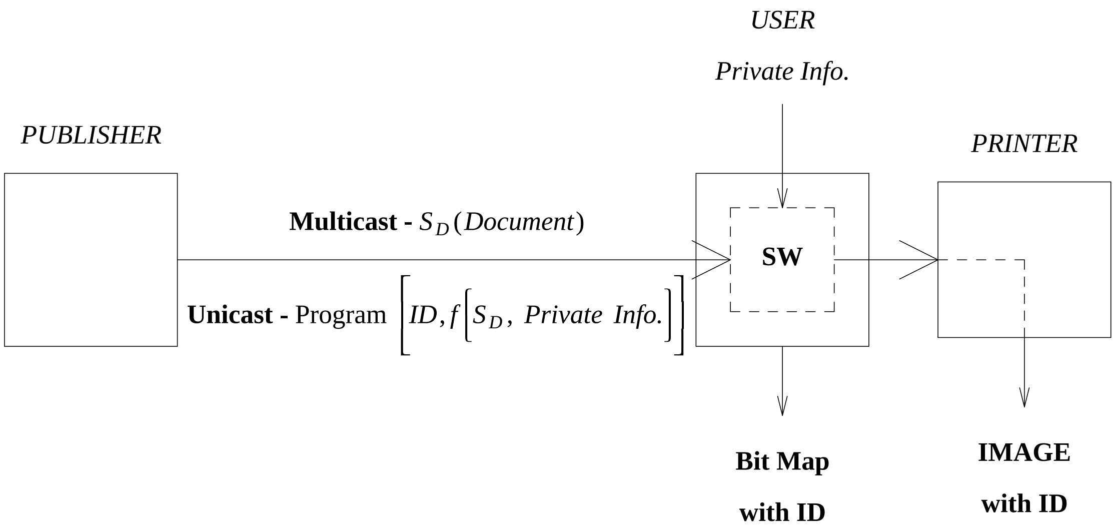


Figure 10. A Document Marking distribution System

Programs can be reverse engineered to extract the secret key, and obtain decrypted versions of an unmarked document. To reduce the value of doing this work, the publisher must change the key and program frequently. Depending upon the value of the documents, the publisher may not send a new copy of the program with every document, but may use the same key for a period of time. For instance, if the publisher distributes a daily newspaper, and the recipients pay by the month, the publisher may elect to change the secret key and program each month. This has the added function that recipients that haven't paid their bills automatically stop receiving the newspaper.

3.3 Privacy

Marking documents and registering the document with the recipient creates privacy issues, particularly in a pull system in which the recipient must request each article that interests him. In this environment the publisher not only knows a customer's magazine subscriptions, but also knows the particular articles that were of interest. It is especially important to hide this information in a business environment. In a system that registers an individual's articles, a business competitor may not only determine another company's employee's journal subscriptions, but may also determine that a large number of that company's employees have suddenly become interested in particular technology.

In future networks with abundant bandwidth, the information on specific articles can be hidden by having the recipients request an entire journal and filter out the unwanted articles. In the present Internet, the time needed to transmit the requested information is an important consideration. The anonymous credit card[10] provides a means of hiding the information that an individual received an article unless the article is redistributed illegally.

The anonymous credit card keeps information on an individual's purchases private by spreading information across the network. A credit card company must know an individual's identity in order to extend credit. A merchant knows what his customer is purchasing. However, if the merchant is convinced that he has been paid, there is no need for the merchant to know the customer's identity. Similarly, if the credit card company is convinced that it has been authorized to transfer money on an individual's behalf, it does not need to know the reason for the funds transfer. Of course, there is always the possibility that the funds were fraudulently spent or that the purchase was used illegally. In these instances, the anonymous credit card provides a means to bring the information on identity and purchases together.

The anonymous credit card uses a double locked box protocol. Funds transfers between two specific accounts in two different banks are performed with the double locked box protocol and an intermediary. A customer identifies himself to his bank and presents an encrypted message, the double locked box, with instructions to transfer funds from his account to the account specified in the box. The message can only be decrypted by the intermediary. The bank sends a funds transfer message to the intermediary, including the encrypted message from the customer. The bank's message is signed and numbered so that the intermediary is certain that it is from a trusted bank and that it is not a duplicate. When the intermediary decrypts the message from the customer it obtains the identity of the destination bank and another encrypted message, the box within the box, that can only be decrypted by the destination bank.

The intermediary sends a funds transfer message to the destination bank, along with the encrypted message. The message from the intermediary is also signed and numbered to guarantee that it is from the intermediary and that it is not a duplicate. When the destination bank decrypts the enclosed message it learns the account number to deposit the funds.

Neither bank knows the other bank or the account in that bank. The intermediary knows both banks but does not know the account in either bank. In order to link the source and destination accounts all three entities must provide information. The unique message numbers provide a means of bringing together the information associated with a particular funds transfer.

The complete anonymous credit card system consists of several different types of funds transfers for spending credits and paying debts. In addition to unique messages, information may be brought together through common information. For instance, if both banks associate your social security number with your account, this information can be used for the receiving bank to learn your identity without involving the intermediary. In order to determine how many entities must cooperate in order to join information, a collusion analysis has been developed that includes all of the messages and information in a system[11]. According to the analysis, five separate entities must cooperate in the complete anonymous credit card in order to join a person's identity and purchases. The analysis also shows the collusion path that specifies the entities and their common information.

The assumption in the anonymous credit card is that an organization may be corrupted, no matter how well intentioned it may be, and a program that implements a policy may contain a bug, no matter how well tested it may be. Therefore, the more entities that have to be compromised to join information, the better the system preserves privacy. If this were not the case, we could just find a credit card company that we trust not to release information.

In the electronic publishing system, privacy is maintained by having recipients pay for articles or journals with an anonymous credit card. Instead of associating a marked article with an individual, the publisher associates the article with the unique message number that was used to pay for the article. If unauthorized copies of the article are retrieved, the publisher obtains a subpoena from a law enforcement agency and forces the parties involved in the funds transfer to collude and identify the individual associated with the message number.

In the document marking context, the ability to determine a person's identity when an illegal act has occurred makes it possible to hide the identity of all of the users who operate legally. Even when a person has committed an illegal act his identity is determined without disclosing his entire reading profile.

The anonymous credit card is not sufficient to guarantee a distance of five entities between an article and the recipient's identity in the electronic publishing system. The complete system includes a message path to deliver articles between the publisher and the individual. In the original analysis of the anonymous credit card an individual purchases merchandise and carries it away. The delivery path must be included in the collusion analysis.

If the user is a workstation with a permanent address on the Internet, the publisher knows the recipients identity. The collusion distance is one, the publisher. Mail forwarders[12] have been used to provide anonymity for electronic mail. The forwarder acts an agent for the recipient. The publisher sends the article to the forwarder, who sends it to the recipient. The collusion distance is increased to two. A series of intermediaries can make the collusion distance as high as needed.

If the user is connected to the Internet by phoning an Internet service provider (ISP) then the ISP owns the destination address that is used by the publisher. If the ISP keeps track of which users are connected to the address, then the collusion distance is two. The collusion distance increases if the individual pays the ISP with an anonymous credit card. In this case, the telephone company is in the collusion path, since it can identify the individual at the other end of a connection.

If articles are delivered over a CATV network, and a recipient picks off articles by filtering out all of the articles that he does not want, then this collusion path for message delivery is broken. The CATV company only knows that the article was delivered to one of its customers, without knowing which one.

Thus far we have adapted the anonymous credit card to separate a user’s identity from the articles that he receives. The principles of information separation and collusion analysis can also be applied directly to electronic publishing without using the anonymous credit card. For instance in the system that we have proposed the publisher provides the program, the article, the decryption keys and the marks. In addition, the recipient pays the publisher directly. Instead, a publisher may sell programs that introduce unique marks, the keys (or portions of the keys) to decrypt articles, and credits to purchase articles in bulk, to different retailers.

A person may order a document from the publisher and pay for it with a credit that he purchased from a retailer. The received document may be encoded with two or more keys. (Many encryption procedures commute so that multiple encodings and decodings can be performed in any order.) The recipient buys each of the keys, and an accompanying mark, from different merchants. A merchants associates a mark with a customer, however, he sells the same mark to many customers so that it is difficult to determine a document recipient with only one merchant’s information. Each merchant sells enough different marks that it is unlikely that all of vendors from which keys are purchased will sell the same marks to the same customer. Therefore, several vendors must collude to associate a person’s identity and articles.

In order to associate an individual with an article, one of the parties in the system must know the individual’s identity. Document marking cannot be used in an electronic cash system with complete anonymity. A collusion analysis, in addition to guaranteeing a minimum path length, also guarantees that there is at least one collusion path. In electronic publishing we may want to guarantee that there is more than one independent collusion path. While it is important to protect the privacy of legal users, it is also important to make it difficult for illegal users to hide their identity. When there are independent collusion paths, a user cannot hide his identity by compromising a single unit. If there is only one path. and a user compromises a unit on that path, the path is broken and the user can hide his identity.

4. Experimental Systems

We have implemented a number of prototype systems which use our watermarking techniques to protect documents. This section discusses the implementation of two systems which demonstrate the range of different applications for the technology. In the first system we describe, electronic documents were distributed to a wide audience using the global internet. In the second system, paper correspondence was distributed among a relatively small group of individuals.

4.1 Network Distribution of a Technical Journal

The IEEE Communications Society conducted an experiment using World Wide Web technology to distribute society technical publications. The trial began with the publication of the October, 1995 issue of the *Journal on Selected Areas in Communication*.[13,14] The trial was an immediate success; over 500 readers registered with the service before the paper journal arrived in the first subscriber’s mailbox. More than 2200 readers eventually registered with the system and have downloaded articles from that issue. A novel aspect of this trial was the first use of watermarking technology to protect copyrighted journal articles distributed on the internet.

Each registered user requesting the full text of an article in PostScript format received a uniquely modified version of the article using the line shifting technique described in Section 2. Many readers of technical articles — particularly those containing equations — prefer to read printed rather than displayed documents. Since many distributed articles would likely be printed, it was decided to use line shifting for its robustness, ensuring that the embedded watermark would survive in printed form.

For a web-based, public access document distribution system, it was appropriate to watermark each requested document on a real-time, on-demand basis. Each original PostScript article to be watermarked required a one-time (off line) preprocessing to facilitate encoding. Each system user was required to register to download articles, at which time a unique binary codeword was assigned to the user. When a registered user requested an article, this codeword was used as a key to reposition text. Text lines were repositioned by invoking a Common Gateway Interface (CGI) script written in both Unix *shell* and *awk* languages. A database record was retained to associate each user with their codeword and requested articles.

Implementing watermarking is, in principle, simple. We found that watermarking a PostScript document can be made very fast and consume little computing resources. We chose not to generate and distribute bitmapped versions of the watermarked documents. Printable bitmaps (e.g. PostScript *image* operator) are ideally distributed to make documents both printable and moderately difficult to alter. But distributing images generally lowers presentation quality, increases the size of files to distribute, and requires image rendering before distribution. Unfortunately, rendering is a CPU intensive operation, though there are several system approaches to tackling this obstacle. One approach is to watermark pre-existing images (i.e. bitmaps) directly, as would be necessary for distributing documents existing only in scanned image form. A second alternative is to "look ahead" by encoding and rendering pages before they are requested. Rather than inserting a mark that identifies the recipient when he requests a document, he receives the next available mark and that mark is registered with him.

In contrast to encoders, implementing a decoder to detect corrupted watermarks is technically challenging. Highly reliable decoding performance requires state-of-the-art document analysis tools to perform noise reduction, text "zoning", and word segmentation. Manual intervention by experts can enhance decoding performance in some cases, such as when obvious extraneous writing or smudging appears on a recovered page. Fortunately, decoding is a relatively infrequent operation without a significant time constraint.

By creating this prototype document distribution system, we have demonstrated that our watermarking technique not only works well but is feasible for large scale systems.

4.2 Registered Correspondence

In order to demonstrate the practicality of our marking techniques, the first internal memorandum that we wrote on the topic was marked, using line shift encoding, and registered with the recipients. The marks were included by manually modifying parameters in a postscript file before printing each copy. We challenged the recipients to locate the marks and to return copied or faxed versions of the memorandum for identification. We demonstrated that we could reliably identify the original recipient of copies with marks that were not noticeable.

In many business applications there is a need to register copies of private correspondence. In the past this has been done by stamping a serial number or bar code in the margin. This type of mark not only changes the appearance of the document, but can be covered before making a copy. Our marking techniques do not compromise the aesthetics of the document, and cannot be covered during copying without losing the contents of the document.

As a follow up to the JSAC experiment we constructed a document marking and identification system (DMTS) for use by secretaries at Bell Labs, on an experimental basis. The system incorporated our refinements in marking and detection techniques up to that point. The documents contained duplicate marks in line and word spacing so that they would survive a wider variety of distortion. A person who used the system submitted a word processor version of the document and a list of recipients to the DMTS system. DMTS generated a postscript file and modified parameters to put different marks in each copy that it sent to a printer. Each copy that was printed was preceded by a cover sheet with the name of the intended recipient, and the recipient and mark were entered in a database for future reference. Because the marks were invisible we appended a footnote informing recipients that the documents were marked and registered.

The DMTS system is a reasonable method for creating small numbers, perhaps tens or hundreds, of registered copies of documents. Each copy must be sent to a printer separately. For larger scale applications, like publishing books, the marking mechanism should be incorporated in the processor in a mass printer. There are several ways in which mass printers can be modified to accomplish this. A straightforward way is to construct a machine that generates copies from a bitmap that is derived from a postscript file, an operation that is similar to a single copy postscript printer. As each copy is printed a few parameters in the postscript file are modified and the bitmap is changed slightly.

5. Conclusion

Three techniques have been described for encoding information into text images; (1) line shift encoding, (2) word shift encoding and (3) character modification. The three techniques offer an increasing number of locations for placing information. However, the techniques with the greatest number of locations are least able to survive the distortion introduced by printing, copying and faxing.

Word Shift and line shift encoding can be decoded using (1) the edges of features, (2) the center of mass of features or (3) the correlation between the profiles of distorted image and perfect copies of the various encodings. The three techniques require increasing amounts of processing. Correlation decoding is necessary for word shift encoding when distortion is present, while centroid decoding is adequate for line shift encoding. Decoding based upon edges or baselines is only appropriate for line shift encoding in a low noise environment. The last two decoding techniques require information from the publisher about the original positions of the centroids of lines or the profiles of words. However, because baselines are equally spaced before encoding, this technique has the unique characteristic that information can be extracted from a document without additional information from the publisher. Therefore, information can be extracted by anyone, not only the publisher's agent. Information encoded in line spaces may be extracted by both baseline decoding and centroid decoding depending upon who is extracting the information and how much the document is distorted. There are many instances where a publisher may want to send information along with the document (i.e. keywords for cataloguing or signatures for insuring that the electronic document has not been altered). Since these documents are likely to have remained in electronic form and do not have severe distortion, baseline decoding is appropriate for extracting this information.

Text marks may always be removed by retyping the document. Automatic techniques that perform character recognition on noisy bit maps are continuously improving. We should expect that this technology will eventually be able to remove marks. However removing marks becomes more difficult, and requires more human assistance as documents become richer and more varied. If a document is straight text, with a single font and character size, that is typed in paragraphs across a page, then automatic resetting, even from a noisy bit map, should be nearly perfect. However, when the image is rich in fonts and character sizes, has Greek letters or equations, and includes text in figure captions and figures, resetting the text becomes more complicated and probably requires human intervention to even locate potential marks. Beyond making it more difficult to remove marks, we can make it nearly impossible to substitute one valid mark for another by randomly selecting the marks that we do use from a much larger set of possible marks.

Marking documents makes it necessary for the publisher to create and deliver a different document for every recipient. This is particularly burdensome in applications like daily newspapers where a large number of copies must be delivered in a very short period of time. A system has been developed that uses the multicast facility of the Internet for delivery and client side software, such as Java, to individualize the documents. Another problem with marking is the potential invasion of privacy. The proposed system circumvents this problem with the anonymous credit card. This technology allows the recipients identity to be hidden from the publisher under normal operation because the publisher can recover this information if an illegal act is established.

Document marking has been established as a viable technology for copyright protection in an electronic publishing trial. In October 1995 the IEEE Journal on Selected Areas in Communications was delivered over the WEB. There were over 2,200 recipients who registered for the experiment and received selected articles. Every article that was delivered was marked and registered with the recipient.

REFERENCES

[1] N. R. Wagner, "Fingerprinting," Proceeding of the 1983 Symposium on Security and Privacy, April 25-27, 1983, Oakland, CA, pp. 18-22.

[2] H. S. Baird, "Document Image Defect Models," *Structured Document Image Analysis*, (H. S. Baird, H. Bunke, K. Yamamoto, eds.) Springer-Verlag, Berlin, 1992, pp. 546-556.

[3] L. B. Schein, *Electrophotography and Development Physics*, 2nd Ed., Springer-Verlag, Berlin, 1992.

[4] H. VanTrees, *Detection, Estimation, and Modulation Theory,* vol. 1, John Wiley & Sons, 1968.

[5] S. H. Low, N. F. Maxemchuk, "Performance Comparison of Two text Marking Methods," IEEE JSAC, vol. 16, no. 4, May 1998, pp. 561-572.

[6] S. H. Low, N. F. Maxemchuk, A. M. Lapone, "Document Identification for Copyright Protection Using Centroid Detection," IEEE Trans on Commun., Mar. 1998, vol. 46, no.3, pp 372-381.

[7] J. Brassil, S. Low, N. F. Maxemchuk, L. O’Gorman, "Electronic Marking and Identification Techniques to Discourage Document Copying," IEEE Journal on Sel. Areas in Commun., Oct. 1995, vol. 13, no. 8, pp 1495-1504.

[8] David Flanagan, **Java in a Nutshell,** O’Reilly and Assoc., Inc, 1996.

[9] N. F. Maxemchuk, "Electronic Document Distribution," ATT Technical Journal, Sept. 1994, pg 73-80.

[10] S. Low, N. F. Maxemchuk, S. Paul, "Anonymous Credit Cards," Proceedings of 2nd ACM Conference on Computer-Communications Security, Fairfax, Va. Nov. 2-4, 1994.

[11] S. Low, N. F. Maxemchuk, S. Paul, "Anonymous Credit Cards and Its Collusion Analysis," IEEE Trans. on Networking, dec. 1996, vol. 4, no.6, pp 809-816.

[12] D. L. Chaum, "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms," Proceedings of Communications of the ACM, Vol. 24, No. 2, February 1981, pp. 84-88.

[13] J. Crowcroft, D. Estrin, H. Schulzrinne, M. Schwartz, "Special Issue: The Global Internet," IEEE JSAC, Oct. 1995, vol. 13, no. 8.

[14] J. T. Brassil, A. K. Choudhury, D. M. Kristol, A. M. Lapone, S. Low, N. F. Maxemchuk, L. O’Gorman, "SEPTEMBER: Secure Electronic Publishing Trial," IEEE Communications Magazine, vol. 34, no. 5, May 1996, pp. 48-55.