

Inspecting Money: How to Avoid Negative Bucks

Robert Thibadeau

CMU-RI-TR-87-1

**The Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213**

January 1987

Copyright © 1987 Carnegie Mellon University

**Presented at the International Conference on Machine Inspection of Printed Security Documents,
sponsored by the Bureau of Engraving and Printing, U.S. Bureau of the Treasury, September 1986.**

Table of Contents

1. Introduction	1
2. Rules of the Game	3
3. Principle of Operation	3
4. Scan Resolution	3
4.1. Inspection Speed	4
5. Width of the Camera Channel	4
6. Lighting the Web	5
6.1. Color Requirements	5
7. Programmed Window Correlation	5
7.1. Window Size	9
7.2. Window Shape: Partitioned Correlation	10
7.3. Regression Alpha Weight	11
7.4. Computational Loading	11
8. Explaining Defects	12
9. Summary	12

List of Figures

Figure 1-1:	Results from the CMU PWBIS-II Inspection device. These show the "needle-in-the-haystack" problem.	1
Figure 1-2:	Defect rates independent of size	2
Figure 1-3:	Average defect size	2
Figure 4-1:	Scan of a bill at 300 DPI to illustrate a "pixel" and what a "raster scan order" means.	4
Figure 7-1:	Formulas for normalized correlation and regression	6
Figure 7-2:	An image can be arranged as a matrix of pixels. The values in two corresponding matrices can then be correlated. We will distinguish one matrix, calling it the operator. The other matrix will be called the image.	6
Figure 7-3:	"James A. Baker III" has been copied from another Five and placed on this Five.	7
Figure 7-4:	Now "James A. Baker III" has been moved down to overlap the "James A. Baker III" on this Five.	8
Figure 7-5:	Circular area of detection moving, in raster fashion, over the entire bill as the "region of interest".	9
Figure 7-6:	Difficulties for peak correlation search	10
Figure 7-7:	Large format binary convolver. Now sold by Visual Understanding Systems, Inc., of Pittsburgh.	11

Abstract

The focus of the activity of the Inspection Laboratory in the past several years has been high-speed inspection of patterned surfaces. The Bureau of Engraving prints money on a high-speed "web," which means that large sheets of money are being produced very fast. The question is raised of how to inspect for the print quality at those speeds. People will see either a blur or only a sample of the print. In this report, we use our past experiences to attempt to detail the inspection methods, hardware, etcetera, which would be needed to construct a hypothetical "Inspect-a-buck" system, and the problems that might be encountered along the way.

1. Introduction

It is interesting, and possibly true, that, if a dollar bill has a flaw, the Bureau of Engraving has printed a case of negative money. The dollar is itself craved by some collector, who pays real dollars (certainly more than one) for the defective dollar. To compensate for the real dollars shelved, the Bureau has to print others. Bad money, legally printed, absorbs wealth, where it was supposed to be the vehicle of wealth.

The fact that the U.S. Bureau of Engraving has embarked on printing money on a high speed web raises the question of how to inspect for the print quality at those speeds. People will see either a blur or only a sample of the print. Computers may be no better at those speeds. There are many ways to qualify the inspection job so that the inspection is, in fact, tractable. We could, for example, reduce the inspection problem to a set of physical measurements which cover some large part of the inspection territory and let that suffice. Knowing the right set of measurements to make may add to the delay in getting the equipment up and running. That may not be bad, but there would be a problem if the measurements started involving pattern measurements akin to doing the inspection completely the first time.

The problem is to inspect for the quality of the entire pattern printed on the paper. It is unclear how to define the concept of deviation from perfection. We would like to find an objective way to quantify visual quality. Once we have an objective way to quantify quality, we can go to measurements made by the inspection device. The problem with this scenario is that it is hard to stand behind any objective quantification of visual quality. If the defect is really a silk thread, then that is OK. There is considerable variation in where the print appears, but that is often OK. Small ink gaps can be invisible when they are contextually correct in the dotting. Ink absorption into the paper is very uneven for dollars. And so on.

Units in Square Mils .001 ² "	Defect	Truth No Defect
System Report		
Defect	4,039	3,271
No Defect	1,507	783,858,394
Hit/False Alarm Rates	.73	.00004

Figure 1-1: Results from the CMU PWBIS-II inspection device. These show the "needle-in-the-haystack" problem.

The focus of much of my activity in the past several years has been in high-speed inspection of patterned surfaces. In 1983, my laboratory constructed a device for inspecting printed circuit surfaces that used heuristic rules related to quality control specifications. Some data on the performance of that device are interesting, and, it is important to note, typical of high-speed pattern inspection devices. Figure 1-1 shows the performance of the device on the basis of its main heuristic rule.

In this case there were approximately 5500 square thousandths of an inch worth of defects in almost 800 million square thousandths of surface area. The capability of the heuristic procedure is good as a signal detection procedure. While the hit rate is only 73% the false alarm rate is a mere .004%. These data illustrate good inspection heuristics that effectively address the "needle-in-the-haystack" problem.

The actual performance of this device is likely to be similar to that considered for the inspection of bills. The likelihood of a defect in a dollar bill on the new presses cannot be reliably estimated, but it is not a bad guess to put it at one part in one or two hundred thousand.

The physical size of this "one part" is important. If the size of the part is large, then few bills will be bad, but if the size of the part is small, many bills will be bad. As a matter of principle the size of the part should be the defect size. But defects vary greatly in size. Certainly some web defects might best be characterized as going over a number of bills. Defect probability should be characterized as an integral of the probabilities taken over defect size. The likelihood that a given dollar bill will show a defect is equivalent to the whole integral, which is different from the number we measured empirically. A defect point is not independent of another. If it were, then, since a bill is about 15,000,000 square mils, there would be on average, about 75 point defects on an average bill. Measuring defects per square inch or per square of surface area does not relate directly to the probability that a dollar is bad.

A second source of data was obtained for defects independent of size. The results are shown in figure 1-2.

Units in Defects	Truth	
	Defect	No Defect
System Report		
Defect	69	57
No Defect	62	??
Hit/False Alarm Rates	.52	??

Figure 1-2: Defect rates independent of size

These results show a hit rate of only about 50%. The false alarm rate cannot be precisely estimated: In the other three cases we have a defect which the system saw or the person could see, and these had clear extents to the system or the observer. Nevertheless, if we could do an overall estimate of the defect detection capability, it would still be good because the false alarm rate would surely be small.

Now, we can combine the two sets of results to obtain an estimate of the average defect size. Figure 1-3 shows the ratio of the number of square mils (thousandths of an inch) to the number of defects. The average defect size is on the order of a hundredth of an inch in area, except where the defects were not detected by the heuristic method. There the defects were half the size, or about a quarter of a hundredth of an inch in area. Smaller defects are harder to detect.

Units in Square Mils	Truth	
	Defect	No Defect
System Report		
Defect	58	57
No Defect	24	??

Figure 1-3: Average defect size

All of this is typical behavior for an inspection system operating heuristically to detect defects at high speeds on patterned surfaces.

In this paper I propose we reject heuristic methods and look, if only academically, at the possibility of doing defect detection algorithmically. The main attribute of the method is that it provides us with a standard against which the real-life, heuristic, inspection is done. It allows us to quantify, or at least specify, quality in a way that relates to inspection device performance and human judgement. Existing quality specifications relate only to human judgement.

I can only hope to sketch out the form that such an algorithmic method will have, since the detailed expression of the form represents a considerable amount of work beyond my present resources.

2. Rules of the Game

The rules of the game in Inspection Technology do not abide incomplete formulations. One cannot, for instance, ignore the camera, the electronics structure, and the speed of operation in constructing inspection methods. The methods must be testable as inspection methods even if no one ever chooses to implement the test. This is tantamount to equating a method with a device. Since we can afford to be flippant about naming the device, let us call it the "Inspect-a-Buck" system. I am sure the various vendors will be more prosaic.

There are two things we want out of our algorithmic method: we want to understand what counts as a complete inspection of the bill, and we want to understand something of the computational load on the Inspect-a-Buck system. If we do our job right, this will give us a standard against which further engineering can be measured.

3. Principle of Operation

Most of the dollar does not change its pattern. This implies that the tested pattern can mostly be referenced against an ideal pattern for deviation. Computers hold great promise here because they have unfailing exact memories. Where a person may find it difficult to confirm every detail of a pattern, a computer, if anything, runs the opposite problem of not failing to forget any detail. We must somehow program the computer as to which details count and how they should count. In order to begin to address this question, we have to address several others first.

4. Scan Resolution

The first question to be asked is how finely to represent the pattern. The scans shown in figure 4-1 are at a relatively coarse resolution of about 300 dots to the inch. In image processing terms, the picture element, or pixel, is 3.33 thousandths of an inch square. If we are to see the bill as well as a person can see it with the unaided eye, then we should allow about a thousandth of an inch (28 microns) per pixel, or 1000 dots to the inch. Furthermore, we should also allow that there be about four gray scale levels (or two binary bits of information) to each pixel.

Pixel values are acquired in a raster acquisition or raster scan of the image plane. An image is then a raster, typically left to right, top to bottom, composition of pixel values on a plane as suggested in figure 4-1. Most every camera which is made performs a raster scan to accumulate pixel values.

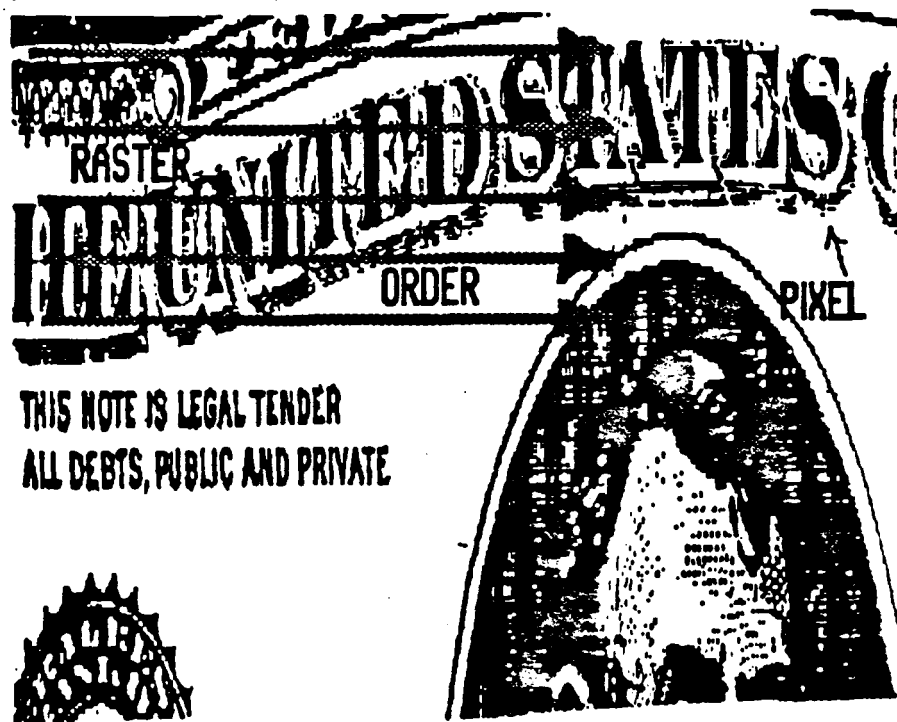


Figure 4-1: Scan of a bill at 300 DPI to illustrate a "pixel" and what a "raster scan order" means.

4.1. Inspection Speed

Speed is a real problem. The web travels at perhaps 300 feet per minute, or 60 inches a second. Say the web is 24 inches across. This provides for 24000 pixels per scan and 1000 scans an inch, or $1000 * 60 * 24000 = 1,440,000,000$ pixels per second throughput. Solid state camera speeds are practical at 15,000,000 pixels per second, which implies we require about 100 independent camera channels operating in parallel on one web.

The Inspect-a-Buck system must operate as fast as this camera data is acquired. The highest speed inspection devices are limited only the raster camera speed. The Inspect-a-Buck system requires multiple independent cameras and camera channels operating in parallel. But each camera will still obtain a raster image of the part of the surface that it is mounted to view. In order to estimate the computational loading on our system, we will have to theorize for a single channel, and then multiply our results by the number of channels, which is, so far, 100.

5. Width of the Camera Channel

One hundred cameras stretched over 24,000 pixels is 240 pixels per camera, or about .24 inches per camera. We are supposing that our cameras are so-called "line scan" cameras which arrange their sensors in a single line, 240 pixels long. The web moving orthogonal to the line forms the image. But each camera must then take in $1000 * 60$, or 60,000 lines of image data per second.

6. Lighting the Web

Freeze action strobes are available which can light the bills for a microsecond, but, unfortunately, they cannot be "refreshed" very fast. Certainly not 60,000 times a second. Let us say we simply go with constant light. Then the "smear" in 1/60,000 of a second is 1 mil. This is probably not excessive in Inspect-a-Buck, but we must anticipate that because the paper is moving as the camera data is read off it in a raster fashion, the images will appear to be tilted slightly. If this tilt is compensated in matching the patterns then it should not be a problem.

6.1. Color Requirements

Since bills are printed with either a uniform black or uniform green ink, there may be little reason to provide color camera data. But, American dollars are printed on paper which is patterned with small imperfections and also colored silk threads.

Ink on paper is subject not only to the intensity of the signal, but also to the saturation of the white paper through the black or green ink, it may be important to inspect with color cameras.

A color camera will produce a "red, green, blue" signals corresponding to the three color filtering scheme which is standard for colorimetric imaging. The three color attributes are hue, saturation, and intensity. Saturation refers to the whitening of a color independent of intensity. In the digital world, if we assume S bit or $s = 2^S - 1$ levels grayscale for each color, the equations for hue, saturation, and intensity are:

(Hue calculates the dominant wavelength.)

$$\text{Hue} = \arccos(1/2)(R-G)/(R-B) / \sqrt{((R-G)(R-G)(R-B)(G-B))}$$

If $B > G$ then $\text{Hue} = 2\pi - \text{Hue}$

$$\text{Hue} := \text{round}(s * \text{Hue}) \bmod \text{round}(s * 2\pi)$$

(Saturation is the minimum common value, or the "white" value).

$$\text{Saturation} = \text{round}(s * (1 - 3 * \text{Min}(R, G, B) / (R + G + B)))$$

(Intensity is the average R, G, B amplitude value).

$$\text{Intensity} = \text{round}((R + G + B) / 3)$$

To illuminate colored silk threads, and color in general, we need only view the saturation surface, not the intensity surface. This implies that we require a two, and probably, three, color camera scheme. We now have a need for 300, not 100 camera channels. However, if we only inspect the saturation surface, we only require 100 inspection processing channels for our 300 camera channels.

7. Programmed Window Correlation

Once the image data is passed along a channel, it must then be interpreted with respect to the pattern. The mathematical concepts of regression and correlation will be used in this study because they allow a linear quantification of how closely two patterns match. Figure 7-1 gives the general formulas for normalized correlations and regressions taken over pixel values as shown in figure 7-2.

The fact that a reference "operator" and a target "image" (see figure 7-2) can be correlated to determine the degree of the match between a reference image and a target has been known for years. In fact several "machine vision" systems almost exclusively use this technique with great effect.

Normalized Correlation

$$r_{xy} = \frac{\sum_i (x_i - M_x)(y_i - M_y)}{N S_x S_y}$$

Standard Deviation

$$S_x = \sqrt{\frac{\sum_i (x_i - M_x)^2}{N}}$$

Linear Regression

$$Y = bX + a$$



$$b = \frac{r_{xy} S_y}{S_x} \quad \text{a solved by means } M_x, M_y$$

Figure 7-1: Formulas for normalized correlation and regression

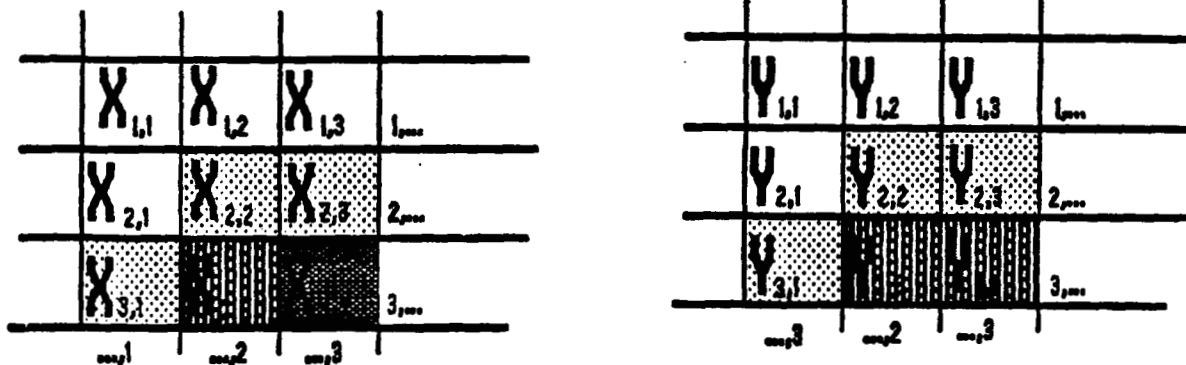


Figure 7-2: An image can be arranged as a matrix of pixels. The values in two corresponding matrices can then be correlated. We will distinguish one matrix, calling it the operator. The other matrix will be called the image.

The Bureau of Engraving underwrote a device some years ago which computed a correlation between an operator the size of the bill and the entire image of the bill. That method failed to detect localized, small, defects, since a high degree of match in one part of the bill could hide a poor degree of match in another part of the bill.

A big problem with such a global matching scheme is shown as we attempt to overlay "James A. Baker III" on the Five as shown in figures 7-3 and 7-4. Note that while the match is very good around "...mes A..." it is poor elsewhere. There is very little dimensional stability in the Five.

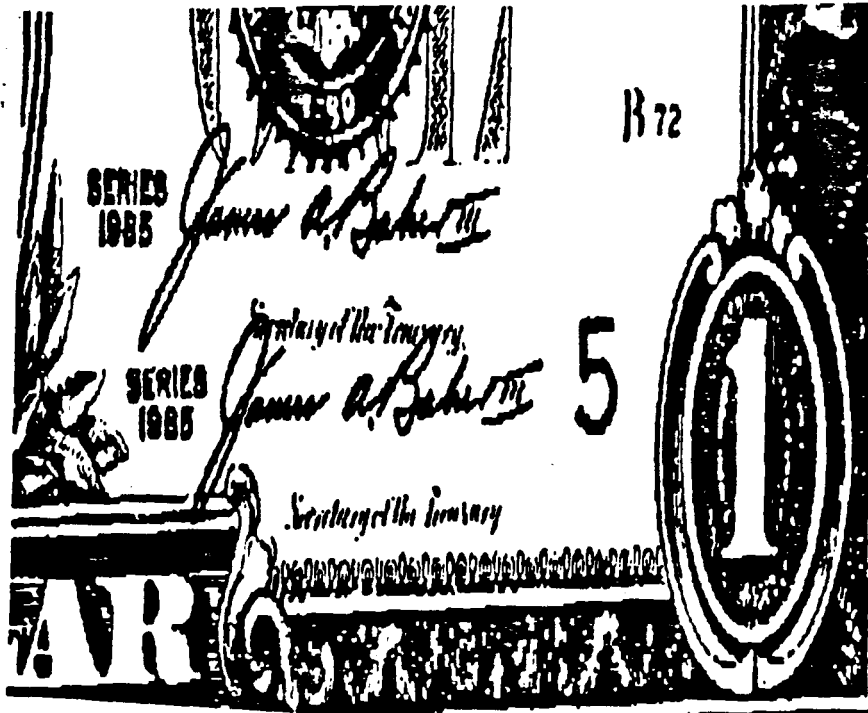


Figure 7-3: "James A. Baker III" has been copied from another Five and placed on this Five.

The dimensional instability demonstrated in figure 7-4 is the rule and not the exception. While people can verify patterns well, people have a very hard time seeing such smooth changes in dimensions. For example, you may be surprised when you measure an even grid on your TV set that the grid may be out as much as an inch or two over 10 or so inches! Unfortunately, while this dimensional instability does not affect people's judgement of quality, it is disaster for a simple correlation scheme.

Inspect-a-Buck uses many small operators, each targeted to a particular part of the dollar bill. Many correlations will then be computed, each corresponding to the inspection of some small section of the bill. This is illustrated as a "circular area of detection" in figure 7-5. Imagine, if you will, this circular area of detection moving over a section of the bill. It may have an operator to detect the "5" in "1985". When the correlation goes high, you have found the "5", and you have confirmed its appearance. In other words, the "5" is both found and inspected. If you never get a sufficiently high correlation on "5" to recognize it, then there is a problem with the bill and it is located where the "5" should have been. I believe this is about as good as the inspection can be.

In order to inspect a bill, Inspect-a-Buck will have to perform many correlations of many operators over the bill's image. The work load can be reduced by having a rough idea of where the operator should

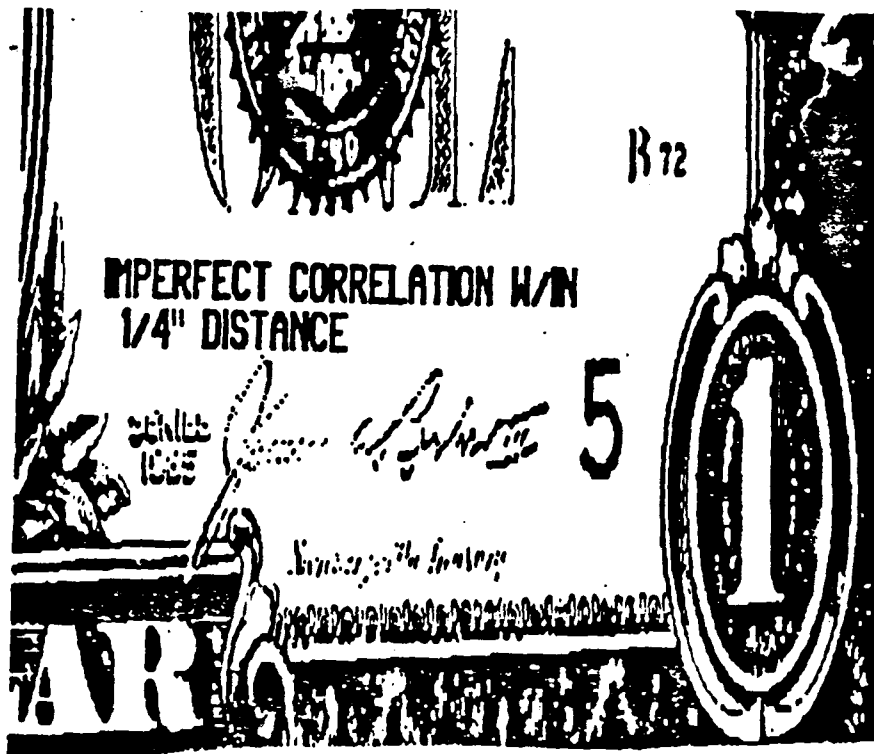


Figure 7-4: Now "James A. Baker III" has been moved down to overlap the "James A. Baker III" on this Five.

apply. We can figure that a photodiode can detect the edge of the print with precision and thereby keep the search region down to few hundredths of an inch. In fact, since the operators themselves achieve register, we can use the first operators to constrain where following operators have to look.

Each operator is associated with a particular part of the bill. In the case of serial numbers, an operator may be associated with a number of "nodal" locations across a large area. Many times the object of the correlation process is to identify a peak correlation value in order to determine (a) the peak's existence, (b) where it is, and (c) whether it is high enough to count as a non-defective feature. The nice feature of the normalized correlation is that it can be used in statistical inference: the system can train on good bills and then later show rejection of bad ones.

The idea of finding a peak in the correlation surface which results from moving a correlation operator over an image segment is not complete. Figure 7-6 illustrates the four major exceptions for bills. The changing patterns associated with serial numbers are handled using the notion of alternative peaks or an operator having several possible nodal locations. The repeating patterns which are prevalent at small scale on the dollar will cause multiple peak correlations. Vertical lines, if the operator is also, by chance, a vertical line, will cause a vertical ridge in the correlation surface, with no peak. Finally, a smooth area with a smooth area for the operator will leave a flat, undefined, correlation (since the variance in X and Y is 0). If we look in the limit as the variance goes to 0, we still have prediction ($Y = a$), so let this correlation be 1.0 by definition. If there is a defect in the smooth area, the correlation will go below 1.0. In this case, then, a dip in the surface will indicate a defect.

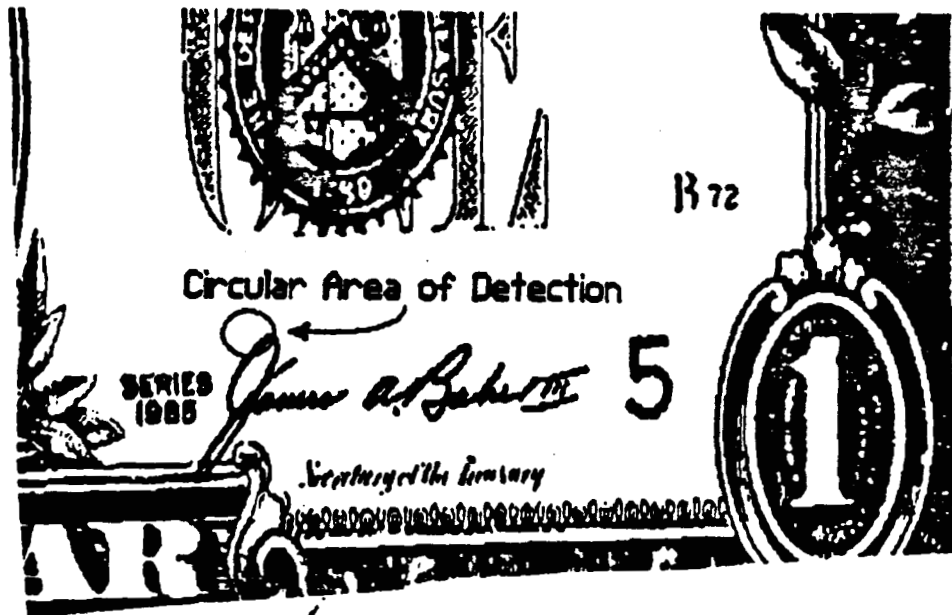


Figure 7-5: Circular area of detection moving, in raster fashion, over the entire bill as the "region of interest".

7.1. Window Size

The size of the windowing operator is a major question in the Inspect-a-Buck system. There are ways of deciding optimal sizes for given patterns, essentially by hill-climbing on the effectiveness of the operator in distinguishing good from bad. A operator which is too small will fail to distinguish because it is overly sensitive to small blemishes, while an operator which is too large will fail because of the large scale dimensional instability or because of its relative insensitivity to small blemishes. An experiment could be constructed to determine optimal window size by comparison with human quality judgements. For the moment we will assume that an optimal size is something large enough to contain an interesting pattern but small enough to clearly avoid the large scale dimensional instabilities. The Inspect-a-Buck system uses a 64 X 64 pixel operator which gives the operator about .064 inch square of coverage.

This operator size affects the number of inspection processing channels which will be required. If a given operator is run over a finite image, say 240 X 240 pixels, then the edge 32 pixels around the image will not get a correlation response since the 64 X 64 operator can only go to the edge (not beyond it). In fact, we only need to move the operator in the region of spatial uncertainty. Let us assume that region can be held to about .1 inch, or, for convenience, 128 pixels. The string of 256 pixel wide camera data needs to be "re-strung" into 256 pixel wide images. Furthermore, a separate 256 wide image needs to be created for each 64 pixel wide spatial operator that we use. Since the web image width is 24,000 pixels, we need to form a minimum of 375 correlation channels of 256 pixel wide data.

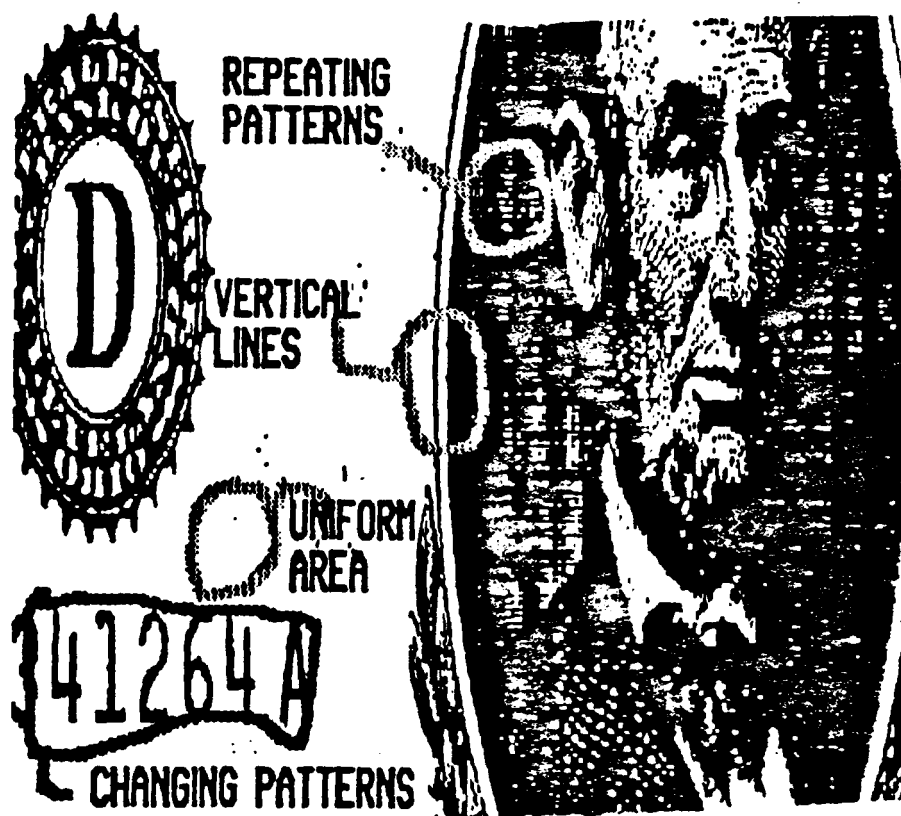


Figure 7-6: Difficulties for peak correlation search

7.2. Window Shape: Partitioned Correlation

Another facet of print quality estimation has to do with whether the print is imperfect or the paper is imperfect. When the print is on top of an imperfection in the paper, the imperfection cannot be seen. Furthermore, it is generally true that an imperfection in the print is more noticeable than an imperfection in the paper. If we imagine a vertical line, the line must be good, but there can be some bleeding into the paper and there may be some lightness variation in the paper which is not allowed in the print.

This suggests that there is a small "don't care" region along the edge between the print and the paper. Furthermore, the dark print area must be essentially perfect, while the light paper area need not be.

In our laboratory, we have demonstrated a machine we call the "large format binary convolver" and numerically related the computation of this machine to "gray scale convolution" and thereby "correlation". That machine permits us to independently correlate the background and the foreground in the image. It furthermore allows us to construct arbitrary "don't care" areas in the operator: areas which do not contribute to the correlation one way or the other. The binary construction also provides a factor of about 10 in cost reduction over multiplier-based circuits for equivalent operations.

7.3. Regression Alpha Weight

Having computed the values needed for a correlation we can also compute the regression equation (see above). This equation includes a number for the "y intercept" or, put in another way, the constant difference between the operator values and the image values. This measures the overall saturation of the ink or the paper or both. Inspect-a-Buck will monitor the difference in saturation from the standard and report if any area is "printed too light or too dark" or the "paper is too light or too dark".

7.4. Computational Loading

Inspect-a-Buck is going to be a very complicated machine. For the sake of comparison I will design it as a composition of "large format binary convolvers". Figure 7-7 shows a picture of one convolver. Each convolver correlates a 64 X 64 Operator with an image area up to 2048 X <infinity> pixels. In Inspect-a-Buck a convolver need only correlate a single camera channel or 240 X 240 pixels. Nevertheless, in terms of pixel operation rate, the convolver operates at better than 40 Billion pixel operations per second. Let us now compute how many convolvers need to be put into Inspect-a-Buck.

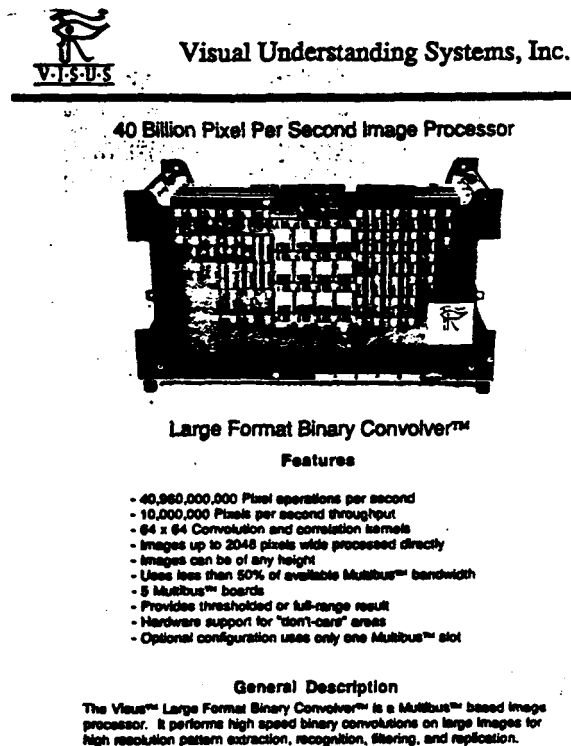


Figure 7-7: Large format binary convolver. Now sold by Visual Understanding Systems, Inc., of Pittsburgh.

The first thing to note is that we allowed that the image have four gray levels, or two bits of information. We we assume that the convolution operator should also have two bits per pixel, then our published proofs show that this will require four convolvers operating in parallel. This means we have 4×375 channels, or 1500 convolvers.

Now, how do we change the operators to select a new operator for each region of interest? We have 375 distinct channels horizontally across the web to handle the 375 distinct 64 pixel wide regions. What about the vertical regions? If we assume a bill is about 3 inches high, that gives 3000 lines, or about $3000/64 = 47$ distinct correlations. This can be handled by arranging convolvers in pairs (possibly triplets, if the timing does not work out). The first convolver will be moving the operator over the image while the second is loading a new operator. Then when the first part of the image is done, the second picks up on the second part with the second operator. The procedure exchanges back and forth continuously. This now doubles the number of convolvers we need, to $2 * 1500$ or 3000.

This should pretty much complete the problem. We can now move 64 X 64 operators in 256 X 256 Fields in an appropriate way over the entire web surface at .001 inch grayscale resolution. We tune each operator to function correctly. Let us add about 200 convolvers to handle the tricky issue of changing print (assuming we inspect after serialization). We can also subtract some for the uniform white areas, if we are careful. The Inspect-a-Buck operation rate is about 15,000,000 (camera speed) X 4096 (operator size) X 3200 (number of distinct convolvers) or 196,608,000,000,000 pixel operations a second (200 quadrillion pixels a second). The poor Cray computers do not reach this. Inspect-a-Buck may have a silly name, but it will be an awesome device.

8. Explaining Defects

Inspect-a-Buck only detects defects, it does not classify them. For some years we have argued that the appropriate way to go about high speed pattern inspection is to do defect detection followed by defect interpretation. We stand by that rule in Inspect-a-Buck. Defect interpretation is almost always a fairly easy thing to do if there is time in software to do it. By filtering image data for interest value (in defect detection), software can be written which will go a long way in explaining and classifying defects as they are found. The rule is simple, though hard to follow: keep a copy of the image that suggested the defect around for the software to analyse. Do not think the defect descriptors available from the defect detection machinery will be sufficient.

9. Summary

The Inspect-a-Buck system is hypothetical but it does help illustrate the scope of the problem of inspecting dollar bills. There seems little doubt that the engineering to be applied in the next few years will yield many new insights into the inspection of money. There is also a very high likelihood of "spin off" technologies applied in printing industries and many industrial inspection applications. Perhaps someday we will have a system which can inspect the Persian Rug in the Living Room and apply the right solutions to cleaning it, automatically.