International Symposium on Multimedia Information Processing (ISMIP 99), Taipei, Taiwan, Dec. 1999

Distortion Modeling and Invariant Extraction for Digital Image Print-and-Scan Process

Ching-Yung Lin and Shih-Fu Chang

Department of Electrical Engineering
Columbia University
New York, NY 10027, USA
{cylin, sfchang}@ee.columbia.edu

ABSTRACT

After an image is printed-and-scanned, it is usually filtered, rotated, scaled, cropped, contrast-and-luminance adjusted, as well as distorted by noises. This paper presents models for the print-and-scan process, considering both pixel value distortion and geometric distortion. We show properties of the discretized, rescanned image in both the spatial and frequency domains, then further analyze the changes in the Discrete Fourier Transform (DFT) coefficients. Based on these properties, we show several techniques for extracting invariants from the original and rescanned image, with potential applications in image watermarking and authentication. Preliminary experiments show the validity of the proposed model and the robustness of the invariants.

KEYWORDS: Printing, Scanning, Rotation, Scaling, Cropping, Watermarking

1 Introduction

Today the print-and-scan (PS) process is commonly used for image reproduction and distribution. It is popular to transform images between the electronic digital format and the printed format. The rescanned image may look similar to the original, but may have been distorted during the process. For some image security applications, such as watermarking for copyright protection, users should be able to detect the embedded watermark even if it is printed-and-scanned. In image authentication cases, the rescanned image may be considered as authentic, because it is a reproduction of the original.

Little work has been done to understand the changes that digital images undergo after the PS process. Most work discusses individual models of printing or scanning. In this paper, we begin with the characteristics of the PS process. Then, in Section 3, we propose a model that can be used to analyze the distortion of a discretized digital image after the PS process in the spatial and frequency domain. Then, we will analyze the variations of DFT coefficients, leading to important properties for extracting invariants. In Section 4, we discuss several methods that can be used to extract invariants of the PS process. Some experimental results, including an analysis of the feature vector proposed in [5], are shown in Section 5. In Section 6, we make a summary and discuss some future work.

2 Properties of the print-and-scan process

Distortion occurs in both the pixel values and the geometric boundary of the rescanned image. The distortion of pixel values is caused by (1) the luminance, contrast, gamma correction and chromnance variations, and (2) the blurring of adjacent pixels. These are typical effects of the printer and scanner, and while they are perceptible to the human eye, they affect the visual quality of a rescanned image.

Distortion of the geometric boundary in the PS process is caused by rotation, scaling, and cropping (RSC). Although it does not introduce significant effects on the visual quality, it may introduce considerable changes at the signal level, especially on the DFT coefficients of the rescanned image.

It should be noted that, in general image editing processes, geometric distortion cannot be adequately modeled by the well-known rotation, scaling, and translation (RST) effects, because of the design of today's Graphic User Interface (GUI) for the scanning process. From Figure 1, we can see that users can arbitrarily select a range for the scanned image. We use "cropping" to describe this operation, because the rescanned images are cropped from an area in the preview window, including the printed image and background. The RST model, which has been widely used in pattern recognition, is usually used to model the geometric distortion on the image of an observed object. In those cases, the meaning of RST is based on a fixed window size, which is usually pre-determined by the system. However, in the PS process, the scanned image may cover part of the original picture and/or part of the background, and may have an arbitrarily cropped size. These changes, especially that of image size, will introduce significant changes of the DFT coefficients. Therefore, instead of RST, a RSC model is more appropriate to represent the PS process. We will discuss this in more detail in Section 3.

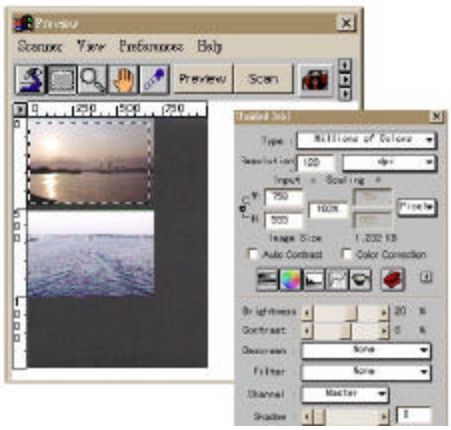

Figure 1: Typical control windows of scanning processes. Users have the freedom to control scanning parameters, as well as can arbitrarily crop the scanned image. [source: Microtek ScanWizard]

3 Modeling of the print-and-scan process

In this section, we first propose a hypothetical model of the pixel value distortions. To our knowledge, there is no existing appropriate model in the literature to describe the pixel value distortions in PS process. Therefore, we propose the following hypothetical model based on our experiments and [1][2]. Although more experiments are needed to verify its validity, we have found this model is appropriate in our experiments using different printers and scanners, as it shows several characteristics of rescanned images. In Section 3.2, we analyze the geometric distortion in the PS process, and then focus on the changes of DFT coefficients for invariants extraction. These models can be applied to general geometric distortions, although a special case (the PS process) is considered here.

3.1 Pixel value distortion

We are interested in modeling the variation of luminance values of color pixels before and after the PS process, because we only use luminance as the main place for embedding information (*e.g.* watermarking) or extracting features in our system. Readers who are interested in color variation can find extensive references in [3]. Our focus is on the popular consumer PS devices such as color inkjet printers and flatbed scanners.

Consumer printers are based on halftoning, which exploits the spatial lowpass characteristics of the human visual system. Color halftone images utilize a large number of small colored dots. Varying the relative positions and areas of the dots produces different colors and luminance values. The lowpass property is usually shown in the spread function of the scanner.

Discrete images are converted to continuous images after printing. In the continuous physical domain, assume we have a virtual finite support image, x, which is reconstructed from the original discrete image, x_0,

$$x(t_1,t_2)=\begin{cases}\sum\sum x_0[n_1,n_2]\,\delta(t_1-n_1T_{01},t_2-n_2T_{02}), & t_1\in[-\frac{T_1}{2},\frac{T_1}{2}] \\ & t_2\in[-\frac{T_2}{2},\frac{T_2}{2}] \\ 0, & \text{elsewhere}\end{cases} \quad (1)$$

where T_{o1} and T_{o2} are the inverse of DPI (dots per inch) values in the t_1 and t_2 directions, and T_1 and T_2 are the range of support of the image. Then, the printed image will be a dithered version of x with additional noises. Combining with scanning process, we assume the pixel value distortion in the PS process can be modeled as

$$x'(t_1,t_2)=K[x(t_1,t_2)*\tau_1(t_1,t_2)+(x(t_1,t_2)*\tau_2(t_1,t_2))\cdot N_1]\cdot s(t_1,t_2) \quad (2)$$

where $x'(t_1,t_2)$ is the output discrete image, K is the responsivity of the detector, and $s(t_1,t_2)$ is the sampling function. There are two components inside the bracket. The first term models the system point spread function,

$$\tau_1(t_1,t_2)=\tau_p(t_1,t_2)*\tau_s(t_1,t_2) \quad (3)$$

where $\tau_p(t_1,t_2)$ is the point spread function of printer, and $\tau_s(t_1,t_2)$ is the detector and optical point spread function of scanner. In the second term, τ_2 is a high-pass filter, which is used to represent the higher noise variance near the edges, and N_1 is a white Gaussian random noise. The noise power is stronger in the moving direction of the carriage in scanner, because the stepped motion jitter introduces random sub-pixel drift. This indicates that τ_2 is not symmetric in both directions

In Eq. (2), the responsivity function, K, satisfies this equation,

$$K(x)=\alpha\cdot(x-\beta_x)^{\gamma}+\beta_K+N_2(x) \quad (4)$$

which includes the combined AC, DC and gamma adjustments in the printer and scanner. N_2 represents that power of noises is a function of pixel value. It includes thermal noises and dark current noises. The variance of N_2 is usually larger on dark pixels, because sensors are less sensitive to their low reflectivity.

From this model, we can analyze the low-pass filtering properties on the Fourier coefficients and describe the high noise variances in the high band coefficients. Some tests of its validity are shown in Section 5.

3.2 Geometric distortion

In general, the scanning process follows a customary procedure. First, a user places a picture (or the printed original image) on the flatbed of the scanner. If the picture is not well placed, this step may introduce a small orientation, or a rotation of 90°, 180° or 270° on the scanned image with a small orientation[1]. Then, the scanner scans the whole flatbed to get a low-resolution preview of the image. After this process, the user selects a cropping window to decide an appropriate range of the

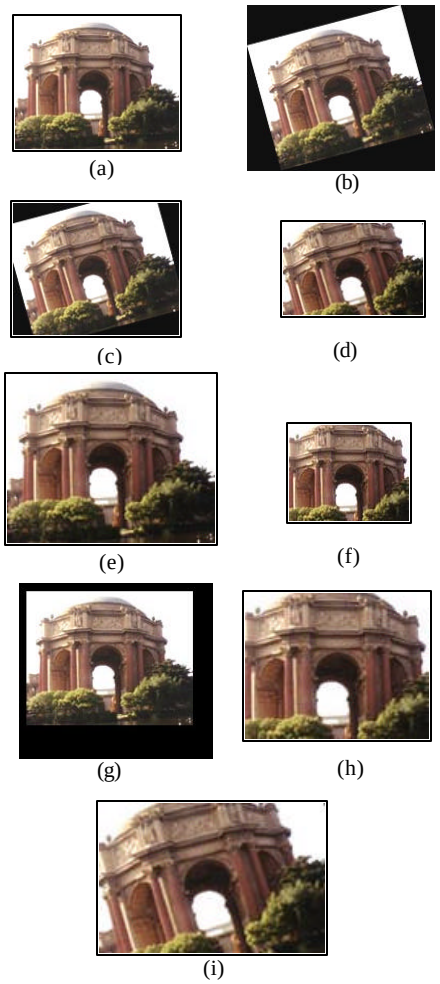

Figure 2: General geometric distortion of images: **(a)** original, **(b)** rotation and cropping with background and the whole image, **(c)** rotation and cropping with background and part of the image, **(d)** rotation and cropping with part of the image, **(e)** scaling, **(f)** cropping without background, **(g)** cropping with background, **(h)** scaling and cropping, and **(i)** rotation, scaling, and cropping.

picture. Usually, it includes only a part of the original image, or the whole picture with additional background (*a.k.a.* zero padding). The scanner then scans the picture again with a higher resolution to get a scanned image. The size of this image is usually different from the original, because the resolution in the scanner and the printer may be different. The final scanned discrete image is obtained by sampling the RSC version of the printing-distorted image with additional scanning noise.

Images are discretized at both ends of the PS process, while they are continuous in the intermediate stages of a printout. We should notice that images are first reconstructed to be continuous, then manipulated, and sampled again. Therefore, a continuous-domain definition of geometric distortions will be more appropriate. Examples of the images after general geometric distortions are shown in Figure 2.

In this section, we propose a general model, including multi-stage RSC in the continuous spatial domain, and discuss how to simplify it. We also show the change of Fourier coefficients after RSC. Since DFT is usually used for frequency-domain analysis of discrete images, we will discuss the impact of RSC in the DFT domain, and then show how to choose an appropriate method to calculate DFT coefficients for invariants extraction.

3.2.1 Continuous-domain models for geometric distortion and the definition of RSC

Considering a general case of the geometric distortion introduced by multiple stages of rotation, scaling, and cropping, the distorted image can be represented as

$$\boldsymbol{x_G} = \mathbf{G}\,\boldsymbol{x} \tag{5}$$

where **G** is the geometric distortion operator. For instance, **G** may equal to **RRSCSRCSRSSC**…, where **R**, **S** and **C**, are the operators of rotation, scaling and cropping, respectively.

We first show the individual effect of RSC. If the image is rotated by $\boldsymbol{q}$ counter-clockwisely, *i.e.*, $\boldsymbol{x_R} = \boldsymbol{R}\,\boldsymbol{x}$, then

$$\begin{aligned} &x_R(t_1,t_2) = x(t_1\cos\boldsymbol{q} - t_2\sin\boldsymbol{q}, t_1\sin\boldsymbol{q} + t_2\cos\boldsymbol{q}) \xleftrightarrow{F} \\ &X(f_1\cos\boldsymbol{q} - f_2\sin\boldsymbol{q}, f_1\sin\boldsymbol{q} + f_2\cos\boldsymbol{q}) = X_R(f_1,f_2) \end{aligned} \tag{6}$$

where X is the Fourier transform of x.

If the original image is scaled by $\boldsymbol{l}_1$ in the t_1-axis and $\boldsymbol{l}_2$ in the t_2-axis, *i.e.*, $\boldsymbol{x_S} = \boldsymbol{S}\,\boldsymbol{x}$, then

$$x_S(t_1,t_2) = x(\tfrac{t_1}{l_1},\tfrac{t_2}{l_2}) \xleftrightarrow{F} X(\boldsymbol{l}_1 f_1, \boldsymbol{l}_2 f_2) = X_S(f_1,f_2) \tag{7}$$

We define cropping as the process that crops the image in a selected area (which may include part of background) at GUI window. Cropping introduces three effects on the image: (1) *translation of the origin point of the image*, (2) *change of the support of image*, and (3) *information loss in the discarded area.* They can be considered as a combination of translation and masking. It is well known that translation introduces only phase shift in the frequency domain. Masking includes the second and the third effects. In the continuous domain, the effect of changing support is not evident, because Fourier transform uses an infinite support, and ignores it. However, in the discrete domain, changing the support of image will change the image size. This results in significant effects on DFT coefficients. We will further discuss it in Section 3.2.2.

[1] In our tests, the small orientation is not common, because pictures or documents are usually placed in the corner of the flatbed. Even if they are not well placed, the rotation angle is generally within a small angle, *e.g.*, ±3.

	Operations in the continuous image domain		
	Scaling	**Cropping**	**Rotation**
Change of Fourier coefficients	Scaling	Phase shift + (Information loss)	Rotation

Table 1: Change of Fourier coefficients after operations in the continuous spatial domain.

Changes of Fourier coefficients introduced by information loss can be considered in two ways. First, the cropped image could be a multiplication of the original image with a masking window, which introduces blurring (with the sinc function) in the Fourier domain. The other method is to consider the cropped image, x_C, as a subtraction of the discarded area, $x_{\bar{C}}$, from the original image, *x*. Then, this equation,

$$| X_C(f_1, f_2) | = | X(f_1, f_2) - X_{\bar{C}}(f_1, f_2) | \qquad (8)$$

represents the cropping effect in the continuous Fourier domain. We find that the second method is a better way to describe the cropping effect.

From Eqs. (6)~(8), we can see that rotation and/or scaling in the spatial domain results in rotation and/or scaling in the frequency domain, respectively, while cropping introduces phase shift and/or information loss. These are shown in Table 1.

Geometric distortion of RSC can also be represented by using coordinate mapping and masking. For instance, a geometric distortion of single rotation, scaling and cropping, sequentially, can be described by

$$\begin{bmatrix} t_1' \\ t_2' \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} t_1 \\ t_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \qquad (9)$$

and

$$x_G = \begin{cases} x', & (t_1, t_2) \in \mathbf{M} \\ 0, & elsewhere \end{cases} \qquad (10)$$

M is a masking function and x' is the image after coordinate mapping. Eqs. (9) and (10) show that RSC can be considered as RST + masking.

How to simplify Eq.(5)? One solution is to reduce multiple RSC operations to a combination of single rotation, scaling, and cropping. First, adjacent similar operations, e.g., **RRR**, can be represented by a single operation. Second, from Eq. (9), we can easily verify that **RC**, **SC** are all inter-changeable. In other words, a rotation operation after cropping can be substituted by a (different) cropping operation after rotation. We notice that **RS** is not inter-changeable unless the scaling factors in t_1 and t_2 dimensions are the same. Therefore, only in the case that images are scaled with the same aspect ratio can Eq. (5) be simplified. Or, Eq.(5) can also be simplified, if rotation is not allowed.

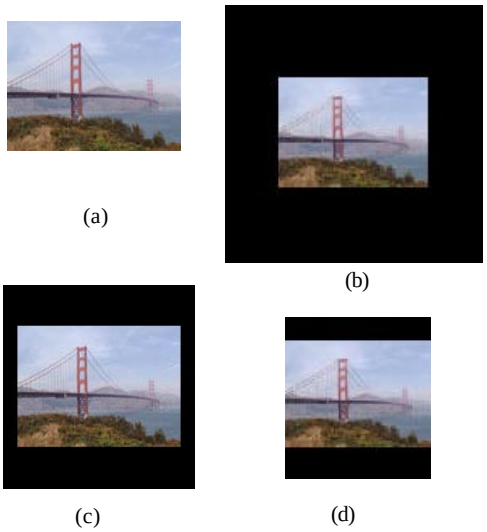

Figure 3: Four common methods to calculate DFT coefficients. The length and width of DFT window are: (a) the image size, (b) a fixed large rectangle, (c) the smallest rectangle with radix-2 width and height, (d) the smallest square including the whole image.

If we only focus on a simple print-and-scan process, then the geometric distortion of the image is a special case of Eq. (5). The manipulations are in the order of rotation, scaling, and cropping. We notice that, without deliberate adjustment, the scaling factor in this process is usually the same in both directions. Therefore, the geometric distortion of PS process in the continuous domain can be described by Eq. (9) with the $\lambda_1 = \lambda_2$. In the continuous Fourier domain, the changes are a combination of Eqs. (6)~(8). Unlike scaling, cropping usually results in a different image size that does not keep the aspect ratio of the original.

3.2.2 Discrete-domain models for geometric distortion

We first define the geometric distortions in the discrete domain. The discretized image is sampled from distorted continuous image, x_G. As we have mentioned, geometric distortion is better described in the continuous domain. Therefore, when we refer to a rotated discrete image, that means the image is converted to the continuous domain, then rotated, and sampled again using the original sampling rate. In practice, discrete images may not be really converted to the continuous domain, but it is possible to use interpolation to approximate this operation. The same definition applies to scaling and cropping. It should be noted that, because using a fixed sampling rate on the scaled continuous image is the same as using a different sampling rate on the original image, "change of sampling rate" and "scaling" indicate the same operation in the discrete-domain models.

It is well known that, in practical implementation, DFT coefficients can be obtained by using radix-2 FFT with zero padding. Some other fast methods of calculating

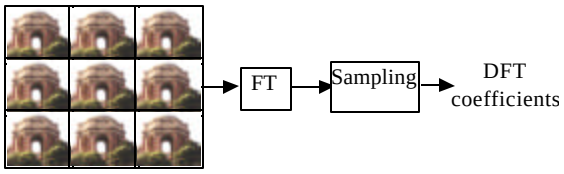

Figure 4: DFT coefficients are obtained from the repeated image.

DFT without using radix-2 FFT are also available. For example, Matlab calculates DFT coefficients by using the original size without zero padding. One of the two methods is usually used for calculating 2-D DFT of the sampled image. They are shown in Figures 3(a) and 3(c). Figures 3(b) and 3(d) show some alternatives mentioned in the literature. All of these methods can be used to obtain DFT coefficients. However, different calculation methods introduce different responses to the coefficients after geometric distortion. Unfortunately, this phenomenon is usually overlooked. In the following paragraphs, we will show some general properties of DFT coefficients, and then analyze them.

· General properties of DFT coefficients

We first show the relationships between continuous Fourier coefficients and DFT. Once a continuous image is discretized, its Fourier coefficients become periodic (and are continuous). They are called the Discrete-Time Fourier Transform (DTFT) coefficients. For images, because their support is finite, we can periodically repeat it in the spatial domain. This will discretizes DTFT coefficients, and gets DFT coefficients. In other words, DFT coefficients are sampled from the Fourier spectrum of the repeated discrete image (see Figure 4). Alternatively, if we first consider the periodicity of an image and then consider its discrete property, DFT coefficients will be the same as Fourier Series (FS) coefficients, with additional noise introduced by aliasing effect.

Figure 5 shows how DFT coefficients change with different spatial sampling rate and different DFT size. Figure 5(a) is a continuous 1D signal and its corresponding Fourier coefficients. This signal is then discretized. The DFT coefficients (DFT window size T_0) of the discretized signal are the samples in the frequency domain. Figure 5(b) shows that the frequency sampling interval ($f_0=1/T_0$) is determined by the repetition period (T_0), *i.e.*, the size of DFT. It is obvious that DFT size plays an important role in the final coefficients. For example, consider the case when the DFT size keeps a fixed ratio to the signal/image size. Then, in Figure 5(c), if the signal is up sampled (or scaled) by 2, we can see that the sampling position of the DFT coefficients in Figure 5(b) and 5(c) are the same, with only difference in the aliasing effect. This is different from the continuous case, where scaling in the continuous domain results in scaling in the continuous Fourier domain. Figure 5(d) shows the effect of zero padding. The more we pad zeroes outside the image, the smaller the sampling interval in the frequency domain will be. Using these properties, we can model the change of DFT coefficients, which are calculated from the four cases in Figure 3, after geometric distortion.

Case I: DFT size equals the image size

In the first case, if the image is scaled, then the FS coefficients of the repeated original continuous image, $\tilde{X}$, and the scaled image, $\tilde{x}_S$, should be the same at the same indices. That is,

$$\begin{aligned}\tilde{X}_S[n_1,n_2] &= X_S\left(\tfrac{n_1}{T_{S1}},\tfrac{n_2}{T_{S2}}\right) = X\left(\tfrac{n_1 l_1}{T_{S1}},\tfrac{n_2 l_2}{T_{S2}}\right) \\ &= X\left(\tfrac{n_1}{T_1},\tfrac{n_2}{T_2}\right) = \tilde{X}[n_1,n_2]\end{aligned} \tag{11}$$

where T_{S1}, T_{S2} are the sizes of the scaled image, and T_1, T_2 are sizes of the original image. Adding the concern of discretization in the spatial domain, we can get the DFT coefficients in the scaled case, $\hat{X}_S$ as

$$\hat{X}_S[n_1,n_2] = \hat{X}[n_1,n_2] + N_{sampling} \tag{12}$$

where $\hat{X}$ is the DFT of original image. Eq. (12) indicates that, after scaling, the DFT coefficients at each indices are still the same as the original with only (sampling) aliasing noises. We can see this property from Figure 5(c). It should be noted that $\hat{X}_S \subset \hat{X}$ or $\hat{X}_S \supset \hat{X}$, because the numbers of sampling points are difference in these two

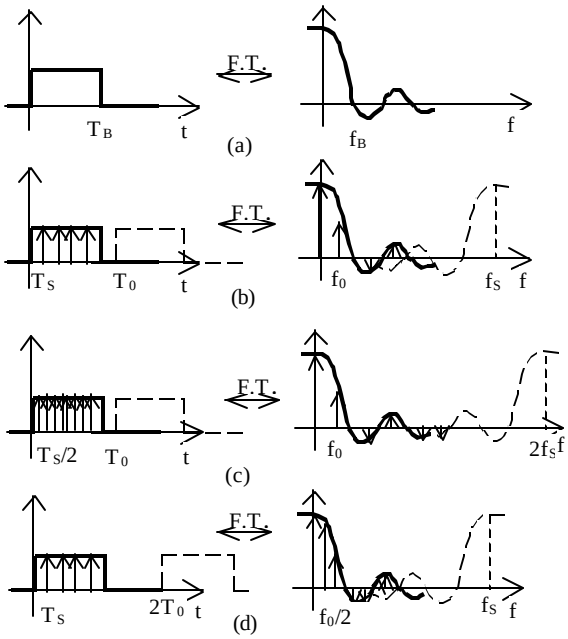

Figure 5: The relationship of DFT coefficients and Fourier coefficients: (a) the original continuous signal, (b) the discretized signal (c) the up-sampled signal (or enlarged signal in a 2-D image), and (d) the zero-padded signal.

images. In Eq. (12), the power of sampling noises is larger, if the image is down-sampled.

In this case, the size of the cropped image will be the DFT size. If we assume this size to be $\alpha_1 T_1$ x $\alpha_2 T_2$, then the DFT coefficients after scaling and cropping are,

$$| \hat{X}_{SC}[n_1,n_2] | = | \hat{X}[\frac{n_1}{a_1},\frac{n_2}{a_2}] + \hat{N}_{SC}[n_1,n_2] | \qquad (13)$$

where

$$\hat{N}_{SC}[n_1, n_2] = -\hat{X}_{\bar{C}}[\frac{n_1}{a_1},\frac{n_2}{a_2}] + N_{sampling}. \qquad (14)$$

In Eq. (13), if the cropped area include the entire original image, i.e., α_1, α_2 >= 1, then the effect of the discarded area can be ignored. If the cropping ratios are too small, then the power loss in the discarded area may not be just ignored as noises. The reliable minimum thresholds that can be considered as noises depend on the system design and specific images. In Eq. (14), strictly speaking, there is no definition in $\hat{X}$ at the non-integer positions. But, since $\hat{X}$ are samples of X, we can set $\hat{X}[\frac{n_1}{a_1},\frac{n_2}{a_2}] = X(\frac{n_1}{a_1 T_1},\frac{n_2}{a_2 T_2})$ directly from the original Fourier coefficients. In practical applications, these values are generally obtained from interpolation.

In cases where DFT size equals image size, rotation in the spatial domain results in the same rotation in the frequency domain.

Several properties of the change of DFT coefficients after geometric distortions are listed in Table 2. In the other three cases, these properties can be readily verified by similar methods in the first case. Thus, we will only discuss them later.

Case II: DFT size is a fixed large rectangle

When calculating DFT, if the number of DCT coefficients is fixed, then the properties of RSC operations are the same in the DFT domain and the continuous Fourier domain. We can see it by comparing Table I and Table II. In this case, previous discussions of the continuous cases are all valid in the DFT domain. However, this method is not practical because it requires a very large fixed-size DFT window for all images.

In cases where DFT size is a fixed large rectangle, Eq. (13) and (14) are still applicable, but α_1 and α_2 should be replaced by λ_1 and λ_2.

Case III: DFT size is the smallest rectangle with radix-2 width and height

The third case in Figure 3(c) is widely used, but it introduces an unpredictable scaling effect, if image sizes change across the boundary of two radix-2 values, (*e.g.*, sizes changed from 127x127 to 129x129). This unpredictable property makes the invariant extraction process more difficult in practical applications. In this case, α_1 and α_2 in Eq. (13) and (14) should be replaced by other more complicated values that are functions of image sizes, scaling factors, and cropping factors.

	Operations in the discrete image domain		
DFT Size	**Scaling**	**Cropping**	**Rotation**
Case I	Almost no effect*	Scaling + Phase shift + (Information loss)	Rotation
Case II	Scaling	Phase shift + (Information loss)	Rotation
Case III	Scaling	Phase shift + (Information loss) + (Scaling)	Rotation
Case IV	Scaling in one dimension and no effect* in the other dimension	Scaling + Phase shift + (Information loss)	Rotation

*: No changes on sampling positions but may introduce different aliasing effect. See Eq. (12).

Table 2: Change of DFT coefficients after operations in the discrete spatial domain.

Case IV: DFT size is the smallest square including the whole image

In this case, since cropping and scaling may also introduce unpredictable scaling effects in the DFT coefficients, similar problems occur as in Case III.

- Rotation

The DFT coefficients of the rotated image have two important properties: the 'cross' effect and the Cartesian sampling points. In Figure 6, we can see that the spectrum of the original image holds a strong cross, which is caused by the discontinuity of pixel values after the image is repeated as in Figure 4. After rotation, if the image includes the whole original and additional background, then this 'cross' will also rotate with the image. However, if the rotated image is cropped as in Figure 6, then this cross will not be rotated, while other coefficients are rotated. In other words, the shape of the support of image decides the angle of 'cross'. We found that this phenomenon becomes less noticeable, if images are subtracted by their mean values before calculating DFT coefficients. Observing from Figure 5(b) and 5(d), we can see the spectrum of a cropping mask The larger the distance of repetition period and the support of mask, the larger the magnitudes of the sidelobe pulses would be. Since these pulses convolve with all the DFT coefficients in the frequency domain, large DFT coefficients domain the values along the angle of the mask. In implementation, we have to notice this effect, and in

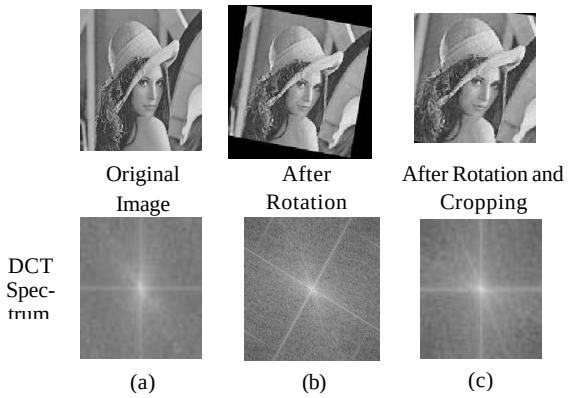

Figure 6: The spectrum of rotated-zero-padded image and rotated-cropped image

acknowledge that some DFT values may be affected near the angle of mask.

DFT coefficients of a discrete rotated image are sampled from the Cartesian grid points of the rotated original continuous spectrum. Therefore, they are not the rotated original DFT coefficients. Two methods can be used to solve this problem in practical cases. The first is to calculate DTFT coefficients at the rotated grid point positions. They are exactly the same sampling points as in the original. However, these calculations are time-consuming. The other method is to interpolate from the DFT coefficients. This method can take advantage of FFT, and can get reasonable results in experiments. To improve the accuracy, zero-padding may be applied to the image to conduct interpolation from denser frequency samples. In implementation, we chose to interpolate coefficients from the magnitudes of DFT coefficients, because phase shifting (introduced by translation) could have significantly changed the complex coefficients.

4 Extracting invariants in the print-and-scan process

- Using scaled images for the DFT-domain analysis

Using DFT as a frequency analysis tool, we can manipulate images *a priori* to make their DFT coefficients more predictable after geometric distortions. Here are two examples. We scale images uniformly or non-uniformly to a standard size (*e.g.*, 256x256), and then apply radix-2 FFT. (In the uniform scaling cases, we may need to pad zeros outside the scaled image.) From Eq. (12), we know that scaling introduces almost no effect on the DFT coefficients, if images are not extensively down-sampled.

As we discussed in Section 3.2.1, uniform scaling can be combined with the original single RSC in the PS process, and it still results in single RSC. Therefore, if both original and distorted images are uniformly scaled to a fixed size before calculating DFT, their DFT coefficients should demonstrate the same properties

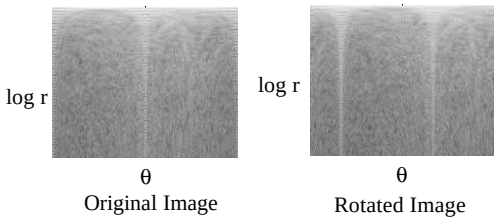

Figure 7: Log-polar map of DFT coefficients. RSC introduces simple shift on this map.

shown in the continuous Fourier domain. Therefore, we can conclude that the DFT coefficients obtained by this method only suffer single rotation, scaling, phase shift, and noises in the PS process. (Here, *scaling* and *phase shifting* in the DFT domain are introduced by *cropping* in the spatial domain, and *noises* are introduced by *scaling* and *cropping* in the spatial domain.)

An alternative method is to non-uniformly scale images to a standard size before calculating DFT. In some applications other than the PS process, such as operations in general image editing software, images may be cropped and scaled with an arbitrary aspect ratio but may not be rotated. This method can be applied to these applications. Examples can be found in [4].

- Using log-polar or log-log map of DFT coefficients to extract invariants

It is well known that the log-polar map of Fourier magnitudes (*a.k.a.* the Fourier-Mellin Transform) possesses simple shifting properties, if images are rotated, translated and uniformly scaled. That is

$$| X_{RST}(\log r, \theta) | = | X(\log r + \log \lambda, \theta + \theta_R) | \qquad (15)$$

where every coordinate point (f_1, f_2) is represented by ($r \cdot \cos\theta$, $r \cdot \sin\theta$). Eq.(15) can be easily verified from Eq.(6) and (7). As we know, the DFT coefficients of a uniformly scaled image have similar properties as in the continuous Fourier coefficients. Therefore, Eq. (15) will be satisfied in the discrete DFT domain. We can use interpolation to obtain the coefficients at log-polar coordinate points. Examples of the log-polar maps of Figures 6(a) and 6(b) are shown in Figure 7.

Since the log-polar map of the rescanned image is a translated version of the original (with noises), it is natural to expect that the 2D DFT magnitudes of this map should be invariant. Therefore, any function of them is expected to be invariant, and served as a feature vector. However, in practical cases, the noises introduced by scanning and cropping are too large to ignore. Also, in the discrete image cases, the invariant-magnitude property of Eq.(15) is valid only if images are cyclically shifted (because DFT is calculated from the repeated image, see Figure 4). This cyclic shift only happens at the axis of θ, but not at the axis of *log* r. Therefore, DFT magnitudes of this map usually do not possess a sufficient invariance.

An alternative method for generating feature vector has been developed in [5], and is summarized as follows. The basic idea is to project all log-polar coefficients along each angle, so that we can obtain a 1D signal that is invariant in the PS process except the cyclic shift introduced by rotation. The feature extraction process is shown in Figure 8. Images are first scaled to a standard size (*e.g.*, 256x256), then zero-padded to double the size (*e.g.*, 512x512). We can get the magnitudes of log-polar coefficients (*a.k.a.* Fourier-Mellin coefficients: *Fm*) from DFT coefficients. The purpose of these steps is to get more accurate |*Fm*|. The next step is to sum up the log values of |*Fm*| along each angle from r_l to r_u, which includes mid-band coefficients. Log values of |*Fm*| are taken so that the summation will not be dominated by principal values, and the utilization of mid-band coefficients is for watermarking. This signal is then divided to two parts, and each value is summed up with the value at its orthogonal direction. There are two reasons. First, the resulted signal will be invariant if the rescanned image is rotated by 90°, 180°, or 270°. Second, its distribution will be more like white Gaussian, which is important to embedding watermark. The final feature vector is the AC component of this signal, which excludes the coefficients near the angle of the axes. This feature vector is very robust. We show some experimental results in Section 5. As mentioned above, rotation introduces cyclic shift to this feature vector. Therefore, in practical PS process, tests should base on shifting the original feature vector within a range, (*e.g.* ± 5°).

In addition to the method in [5], some other methods may be applied to extract invariants. The 1D DFT magnitude of the previous feature vector is an example, which is rotation invariant but less robust. Another example is to use a similar step, but sum up values along each *r* or *log r*. The resulted feature vector will be invariant to non-uniform scaling, rotation, and cropping to the scaled size. As we mentioned, in some cases, non-uniformly scaling and cropping is a more demanding process. We can use the log-log map instead of the log-polar map, because it only suffers simple shifting properties after general scaling and cropping [4].

5 Experiments

• Pixel value Distortion

We tested our models using the EPSON Stylus EX Inkjet printer and the HP Scanjet 4C scanner, both common commercial products. Five different images are tested, and they showed similar results. Here is an example. A color image of 384x256 was printed on the inkjet paper, with the physical size of 5.32"x3.54". Then it was scanned with 75 dpi [size: 402x266]. To isolate the inference of pixel value distortion, we crop, scale, and estimate its sub-pixel translation to register this image to the original. Experimental results are shown in Figures

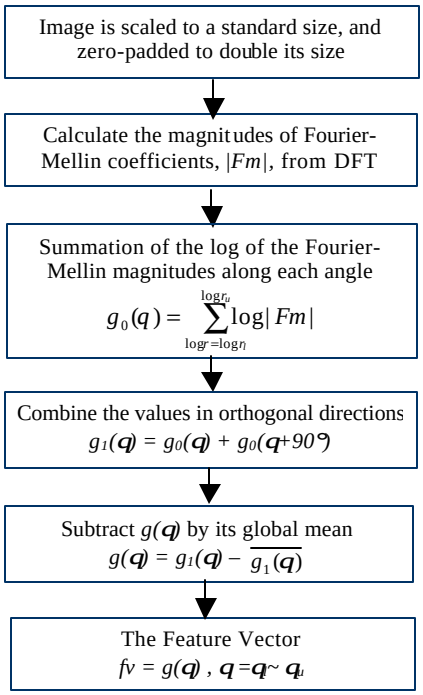

Figure 8: Extract invariants from log-polar map of DCT coefficients.

9(c)~(e). We can see that the noises in the rescanned image are not like additive Gaussian noises. Instead, they depend on pixel values and the spatial distribution of pixels. In Figure 9(c), we show the mapping of the pixel values from the original image and the registered rescanned image. We can see that Eq. (4) can suitably model the distribution of mapping function. We use optimum estimation methods to estimate (α, γ, β_x, β_K), which are (8.3, 0.6, 35, 20). The MSE of estimation noise is 73.58. At Figure 9(d), we show the difference between the original pixels and the gamma-corrected pixel values of the rescanned image. We can see that noises are larger in the edges and the dark areas. The former satisfies N_1 in Eq.(2), and the latter shows N_2 in Eq.(4). In Figure 9(e), we show the difference of the frequency spectrum of original image and gamma-corrected image. We can clearly see the lowpass filtering and high frequency noises in the spectrum.

The above experiment shows the effectiveness of our model of Eq.(2)~(4). In the practical applications, however, if the original image is not available for registration, then we can not estimate the gamma corrected model. In that case, we can use a linear model, *i.e.*, γ=1, in Eq.(4). In Figure 9(c), we can see the result of a linear model, which uses linear regression. The MSE of this model is 124.08. We can see that noises are larger when pixels are very bright or very dark. Noise distribution in the spatial domain is similar to Figure 9(d), but with larger variances in the bright areas Distribution of noises in the frequency domain is similar to Figure 9(e).

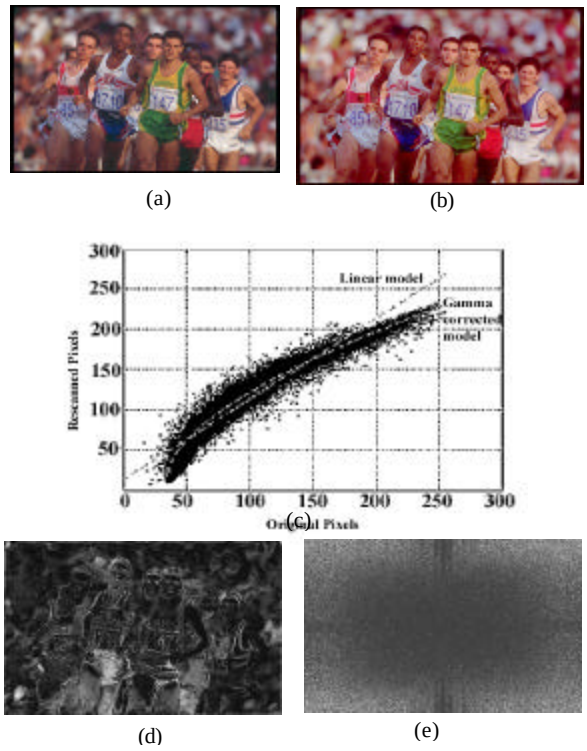

Figure 9: Pixel value distortion of rescanned image. (a) original image [384x256], (b) rescanned image [402x266], (c) corresponding pixel mapping and modeling, (d) noise in the spatial domain after gamma correction, (e) noise in the frequency domain after gamma correction.

We also tested our models by scanning a photo 10 times, and comparing differences. The noise distribution satisfied N_1 in Eq. (2) of our model. Furthermore, we tested some of the cheapest consumer inkjet printers and scanners (which cannot print or scan images higher than 300 dpi), and found their quality is so bad that individual color halftone dots look very distinct in the rescanned image. In these cases, *the rescanned images have to be further blurred by users* to obtain merely acceptable images. Our hypothetical models are found sustainable with a lowpass filter of very low cutoff frequency.

- **Geometric Distortion**

We use the famous Lenna image [512x512] as an example to show the properties of the described feature vector. The experimental results are shown in Figure 10. Correlation coefficients, ρ, of the feature vector extracted from the original image and the distorted image are used to measure the invariance. In our experience, if $\rho>0.6$, then the extracted feature vector will be invariant enough to be applied to public watermarking. It should be noted that no information from the original image is needed for the feature vectors of rescanned image. In these experiments, $(r_l, r_u, \theta_l, \theta_u) = (34, 100, 8°, 81°)$.

In Fig. 10(a), we show that the extracted feature vector is very robust to scaling. Testing is based on different scaling factors, λ, from 0.1 to 2.0. We found that $\rho>0.98$ for all $\lambda>0.25$. In other words, the feature vector extracted from a scaled image which is larger than 128x128 is almost the same as the original. Only if $\lambda<0.12$, i.e., 0.014 of the original area, then the correlation coefficient, ρ, becomes smaller than 0.6.

In Fig. 10(b), we test its robustness against JPEG compression. The testing results are so good that $\rho>0.988$ for all quality factors >30, and $\rho>0.947$ for all quality factors >10.

In Fig. 10(c), the cropped area only includes part of the original and no background (*a.k.a.* strict cropping). We tested three ways: *uniform*, *non-uniform*, and *one-side cropping*. *Uniform cropping* means that the cropping ratios at both axes are the same. We choose cropping factors $\alpha_1=\alpha_2=0.6\sim1$, and show the result by the ratio of cropped area, *i.e.*, $\alpha_1\times\alpha_2$. *Non-uniform cropping* uses different cropping ratios at axes. Their cropping factors are randomly chosen between 0.6~1. The *one-side cropping* method sets $\alpha_2=1$, and α_1 from 0.6~1. These methods result in different image shapes, which affect the feature extraction process. For instance, the largest size, *i.e.*, *max*(width, height), of the distorted image after one-side cropping remains the same as the original. Because images are uniformly scaled to a standard size (256x256) in the feature extraction process, the distorted image will be a subset of the original image. Therefore, only information loss results in the change of DFT coefficients. We can see that information loss is still acceptable if the ratio of cropped area > 0.6. The other two cases introduce scaling in the DFT coefficients, in addition to the information loss. We see that their correlation coefficients are smaller. But, no matter which method is used, a ratio of cropping area > 0.6 is usually acceptable.

In Figure 10(d), we show the test results of general cropping including background. We can see that $\rho<0.6$ if the ratio of cropped area > 2.5. Distortions come from the scaling of DFT coefficients in the extraction process. Although it only introduces shifting in the log-polar map of the DFT coefficients, the loss of coefficients shifted outside the calculated range is too large to ignore. Experimental results are not good in this case. However, this kind of cropping is not common, and we can always crop the image again to obtain a better feature vector.

Figure 10(e) shows the experimental results of rotation. Images are rotated within $\pm$ 3°, and then cropped to the original size. Because the extracted feature vector suffers cyclic shift, tests are based on the largest ρ calculated by shifting the original feature vector in a range of $\pm$ 5°. All ρ values are all acceptable in these cases. To compare the extracted feature vector, we show the results of another method that uses the DFT magnitudes of the feature vector. This method does not

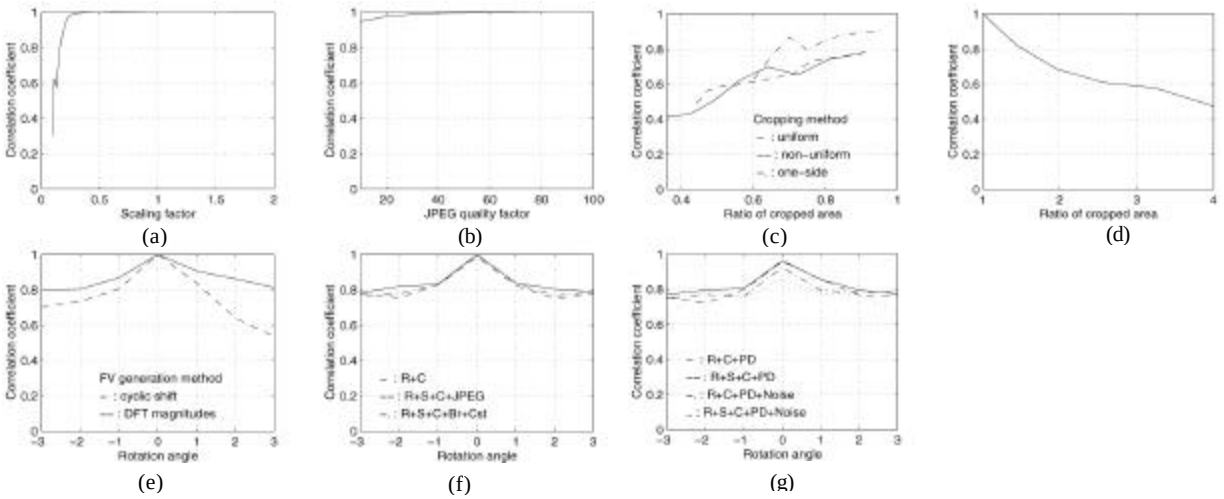

Figure 10: Test of robustness of the extracted feature vector: (a) scaling, (b) JPEG compression, (c) cropping without background (strict cropping), (d) cropping with background (general cropping), (e) rotation with general cropping, and (f) rotation, strict cropping, scaling ($\lambda_1 = \lambda_2 = 0.4$), and JPEG compression (qf = 75) or brightness (β_K=10) & contrast adjustment (α=1.2) (g) RSC, pixel distortion model ($\alpha, \gamma, \beta_x, \beta_K$)=(8.3, 0.6, 35, 20), noise σ^2=74

require any cyclic test of the feature vector, but it is not as robust as the previous method.

- **Geometric distortion + pixel value distortion**

Figure 10(f) shows that the proposed feature vector is *robust to a combined attack of rotation, scaling, cropping, JPEG, brightness and contrast adjustments*. In this test, the image is rotated within ± 3°, strictly cropped with the largest area that does not cover background, scaled with $\lambda_1=\lambda_2=0.4$, and then either JPEG compressed (qf=75) or brightness/contrast adjusted (α=1.2, γ=1, β_x=0, β_K=10). Compared to Figure 10 (e), we can see that distortion of feature vectors are mostly introduced by rotation and cropping, while the effects of scaling, JPEG compression, brightness/contrast adjustments are negligible.

In Figure 10(g), we show the result of a combination of RSC and our pixel value distortion model. The parameters estimated in Figure 9 are used in these experiments. We use an additive Gaussian noise (σ=8.5) in these tests. Because it is distributed in all bands, it will be worse than the real situations in which noises only affect uncalculated high-band. We observed that noises have larger effect in downsized images. Comparing to Figure 10(f), we can see that their results are similar.

We tested the practical rescanned images in Figures 9(a) and (b), and obtained their correlation coefficient, ρ=0.915. Applying the proposed feature vectors for watermarking, we have tested a large database of 20,000 color images from the Corel image library. Their results have proved the invariant properties of the feature vector [5]. Also, a very low false positive rate in those experiments helped prove that the feature vectors from different images are mutually uncorrelated.

6 Summary

Our contribution in this paper is the new, extensive work on modeling the changes that digital images undergo in the print-and-scan process. We propose a model for the pixel value distortion, define the RSC-based geometric distortions, analyze the change of DFT coefficients after geometric distortion, and describe methods to extract invariant feature vector. Preliminary experimental testing of the pixel value distortion, as well as experimental analyses of the feature vector in [5], have indicated the effectiveness of the proposed models. These models can be used in several applications, such as image watermarking[4][5], authentication and registration.

Acknowledgements

Work of the first author was partly supported by the NEC Research Institute. We would like to thank Mr. Yuiman Lui, Dr. Matt Miller, Dr. Jeff Bloom and Dr. Ingemar Cox for sharing ideas and helpful discussions, and Sara Brock for proofreading.

References

[1] X. Feng, J. Newell and R. Triplett, "Noise Measurement Technique for Document Scanners," SPIE vol. 2654 Solid State Sensor Arrays and CCD Cameras, Jan. 1996.

[2] H. Wong, W. Kang, F. Giordano and Y. Yao, "Performance Evaluation of A High-Quality TDI-CCD Color Scanner," SPIE vol. 1656 High-Resolution Sensors and Hybrid Systems, Feb 1992.

[3] G. Sharma and H. Trussell, "Digital Color Imaging," IEEE Trans. on Image Processing, Vol. 6, No.7, July 1997.

[4] C.-Y. Lin, "Public Watermarking Surviving General Scaling and Cropping: An Application for Print-and-Scan Process," Multimedia and Security Workshop at ACM Multimedia 99, Orlando, FL, Oct. 1999.

[5] C.-Y. Lin, M. Wu, M. L. Miller, I. J. Cox, J. Bloom and Y. M. Lui, "Geometric Distortion Resilient Public Watermarking for Images," *SPIE Security and Watermarking of Multimedia Content II*, San Jose, Jan 2000.