Outline

- Introduction
- Resampling
- Detecting resampling
- Experiment results
With the advent of low-cost and high-resolution digital cameras and sophisticated editing software, digital images can be easily manipulated and altered.

Digital forgeries, often leaving no visual clues of having been tampered with, can be indistinguishable from authentic photographs.

As a result, photographs no longer hold the unique stature as a definitive recording of events.
Introduction

• Digital watermarking has been proposed as a means by which an image can be authenticated:
  – embedded signatures
  – robust watermarks
  – fragile watermarks
  – semi-fragile watermarks
  – self embedding watermarks
Introduction

- All of these approaches work by either inserting at the time of recording an imperceptible digital code (a watermark) into the image.
- The major drawback is that a watermark must be inserted at precisely the time of recording.
- This paper describe a technique for detecting traces of digital tampering in the complete absence of any form of digital watermark or signature.
• In order to create a convincing match, it is often necessary to resize, rotate, or stretch portions of the images. This process requires resampling the original image onto a new sampling lattice. It introduces specific correlations into the image, which when detected can be used as evidence of digital tampering.
Resampling

• For purposes of exposition, describe how and where resampling introduces correlations in one-dimensional signals, and how to detect these correlations.
• Consider a 1-D discretely sampled signal $x[t]$ with $m$ samples. The number of samples in this signal can be increased or decreased by a factor $p/q$ to $n$ samples in three steps as follows.

  1) Up-sample: create a new signal $x_u[t]$ with $pm$ samples, where $x_u[pt] = x[t], t=1,2,...,m$, and $x_u[t]=0$ otherwise.

  2) Interpolate: convolve $x_u[t]$ with a low-pass filter:

     $$x_i[t] = x_u[t] * h[t].$$

  3) Down-sample: create a new signal $x_d[t]$ with $n$ samples, where $x_d[t] = x_i[qt], t=1,2,...,n$. Denote the resampled signal as $y[t]=x_d[t]$. 
Resampling

(a) 

(b) 

(c) 

(d)
Resampling

• Since all three steps in the resampling of a signal are linear, this process can be described with a single linear equation.

• Denoting the original and resampled signals in vector form $\vec{x}$ and $\vec{y}$, respectively, resampling takes the form:

$$\vec{y} = A_{p/q} \vec{x}$$

where the $n \times m$ matrix embodies the entire resampling process.
Resampling

- For example,

\[
A_{4/3} = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0.25 & 0.75 & 0 & 0 & 0 \\
0 & 0.50 & 0.50 & 0 & 0 \\
0 & 0 & 0.75 & 0.25 & 0 \\
0 & 0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

\[
A_{2/1} = \begin{bmatrix}
1 & 0 & 0 \\
0.5 & 0.5 & 0 \\
0 & 1 & 0 \\
0 & 0.5 & 0.5 \\
0 & 0 & 1 \\
\end{bmatrix}
\]
Resampling

• Depending on the resampling rate, the resampling process will introduce correlations of varying degrees between neighboring samples.

• For example, consider the up-sampling of a signal by a factor of two using linear interpolation. The odd samples of the resampled signal take on the values of the original signal, the even samples, on the other hand, are the average of adjacent neighbors of the original signal: \( y_{2i} = 0.5x_i + 0.5x_{i+1} \)

• Since each sample of the original signal can be found in the resampled signal, i.e., \( x_i = y_{2i-1} \) and \( x_{i+1} = y_{2i+1} \), the above relationship can be expressed in terms of the resampled samples only: \( y_{2i} = 0.5y_{2i-1} + 0.5y_{2i+1} \)
• That is, across the entire resampled signal, each even sample is precisely the same linear combination of its adjacent two neighbors.

• In this simple case, at least, a resampled signal could be detected (in the absence of noise) by noticing that every other sample is perfectly correlated to its neighbors.
• Consider now resampling a signal by an arbitrary amount $p/q$. In this case, we first ask, when is the $i$th sample of a resampled signal equal to a linear combination of its neighbors, that is

$$y_i \overset{?}{=} \sum_{k=-N}^{N} \alpha_k y_{i+k}$$
• Re-ordering terms, and rewriting the above constraint in terms of the resampling matrix yields

\[ y_i - \sum_{k=-N}^{N} \alpha_k y_{i+k} = 0 \]

\[ (\vec{a}_i \cdot \vec{x}) - \sum_{k=-N}^{N} \alpha_k (\vec{a}_{i+k} \cdot \vec{x}) = 0 \]

\[ \left( \vec{a}_i - \sum_{k=-N}^{N} \alpha_k \vec{a}_{i+k} \right) \cdot \vec{x} = 0 \]

where \( \vec{a}_i \) is the \( i \)th row of the resampling matrix
Resampling

• We see now that the $i_{th}$ sample of a resampled signal is equal to a linear combination of its neighbors when the $i_{th}$ row of the resampling matrix, is equal to a linear combination of the neighboring rows.

• Note also that if the $i_{th}$ sample is a linear combination of its neighbors then the $(i-kp)_{th}$ sample (an integer) will be the same combination of its neighbors, that is, the correlations are periodic.
Detecting Resampling

• Given a signal that has been resampled by a known amount and interpolation method, it is possible to find a set of periodic samples that are correlated in the same way to their neighbors.

• On the other-hand, if we know the specific form of the correlations, then it is straightforward to determine which samples satisfy

\[ y_i = \sum_{k=-N}^{N} \alpha_k y_{i+k} \]
Detecting Resampling

• In practice, of course, neither of the samples are correlated nor are the specific form of the correlations typically known. In order to determine if a signal has been resampled, employ the expectation/maximization algorithm (EM) to simultaneously estimate a set of periodic samples that are correlated to their neighbors, and the specific form of these correlations.
Detecting Resampling

- First assuming that each sample belongs to one of two models. The first model, M1, corresponds to those samples that are correlated to their neighbors, and the second model M2 corresponds to those samples that are not.
- The EM algorithm is a two-step iterative algorithm:
  1) in the E-step, the probability that each sample belongs to each model is estimated.
  2) in the M-step, the specific form of the correlations between samples is estimated.
Detecting Resampling

• More specifically, in the E-step, the probability of each sample \( y_i \) belonging to model M1 can be obtained using Bayes’ rule:

\[
Pr\{y_i \in M_1 | y_i\} = \frac{Pr\{y_i \in M_1 \} Pr\{y_i \in M_1\}}{\sum_{k=1}^{2} Pr\{y_i \in M_k \} Pr\{y_i \in M_k\}}
\]

where the priors \( Pr\{y_i \in M_1\} \) and \( Pr\{y_i \in M_2\} \) are assumed to be equal to 1/2
Also assume that\

\[
\Pr\{y_i \mid y_i \in M_1\} = \frac{1}{\sigma \sqrt{2\pi}} \exp\left[ -\frac{\left(y_i - \sum_{k=-N}^{N} \alpha_k y_{i+k}\right)^2}{2\sigma^2}\right]\
\]

and that \( \Pr\{y_i \mid y_i \in M_2\} \) is uniformly distributed over the range of possible values of the signal.
Detecting Resampling

• In the M-step, a new estimate of $\bar{\alpha}$ is computed using weighted least-squares, that is, minimizing the following quadratic error function:

$$E(\bar{\alpha}) = \sum_i w(i) \left( y_i - \sum_{k=-N}^{N} \alpha_k y_{i+k} \right)^2$$

where the weights $w(i) \equiv \Pr\{y_i \in M_1 | y_i\}$ and $\alpha_0 = 0$.

• Solving for $\bar{\alpha}$ yielding

$$\bar{\alpha} = (Y^T W Y)^{-1} Y^T W \bar{y}$$

where $Y$ is

$$Y = \begin{bmatrix}
y_1 & \cdots & y_N & y_{N+2} & \cdots & y_{2N+1} \\
y_2 & \cdots & y_{N+1} & y_{N+3} & \cdots & y_{2N+2} \\
\vdots & & \vdots & & \vdots & \\
y_i & \cdots & y_{N+i-1} & y_{N+i+1} & \cdots & y_{2N+i} \\
\vdots & & \vdots & & \vdots &
\end{bmatrix}$$

and $W$ is a diagonal matrix with $w(i)$ i along the diagonal.
Detecting Resampling

• example
Detecting Resampling

• There is a range of resampling rates that will not introduce periodic correlations. For example, consider down-sampling by a factor of two (for simplicity, consider the case where there is no interpolation). The resampling matrix, in this case, is given by

\[ A_{1/2} = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 \\
\vdots & & & & \\
\end{bmatrix} \]

• No row can be written as a linear combination of the neighboring rows—in this case, resampling is not detectable.
Experiment Results

- 200 grayscale images with size 512 X 512 pixels in TIFF format.
- Resampled images were up-sampled, down-sampled, rotated, or affine transformed by varying amounts.
- The EM parameters were fixed throughout at

\[ N = 2, \sigma_0 = 0.0075, \text{ and } N_h = 3^2 \]
Experiment Results

$p$  $|\mathcal{F}(p)|$

2.5%  

5%  

10%  

$p$  $|\mathcal{F}(p)|$

2.5%  

5%  

10%
Experiment Results
Experiment Results
Experiment Results

original

forgery

probability map ($p$)
Experiment Results

original

forgery

probability map \((p)\)
Experiment Results
Experiment Results

The graphs show the percentage of correct responses as a function of signal-to-noise ratio (dB). The top graph represents one condition, and the bottom graph represents another. The solid circles indicate consistent performance across different noise levels, while the open circles show a decrease in performance as the noise level increases.
Experiment Results
The reason for the general poor performance of detecting resampling in JPEG compressed images is two-fold.

First, lossy JPEG compression introduces noise into the image. (e.g., a compression quality of 90 introduces, on average, 28 db of noise)

Second, the block artifacts introduced by JPEG introduce very strong periodic patterns that interfere with and mask the periodic patterns introduced by resampling.