Detection of Seam Carving and Localization of Seam Insertions in Digital Images

Anindya Sarkar, Lakshmanan Nataraj and B. S. Manjunath

Vision Research Lab,
Department of Electrical and Computer Engineering,
University of California, Santa Barbara

The 11th ACM Workshop on Multimedia and Security
September 8, 2009
Can You Spot the Points of Change?
Can You Spot the Points of Change?
Figure: row-wise: (a) original image, (b)/(c)/(d) image with 1%/5%,10% seam insertions
**VISUALIZING A SEAM**

**Figure:** Example of seam carving for a 4x5 matrix \( \mathbf{a} \)
Seam Carving/Insertion for Object Removal

**Figure:**
(a) original image  (b) mask shows object to be removed
Seam Carving/Insertion for Object Removal

Figure:
(a) original image (b) mask shows object to be removed, (c) 1 seam removed from region of interest, (d) 45 seams removed, (e) object fully removed through seam carving.
Seam Carving/Insertion for Object Removal

Figure:
(a) 1 seam removed from region of interest, (b) 45 seams removed, (c) object fully removed, (d) selected region after seam insertion with seams shown and (e) without seam insertions shown
**RESULTANT IMAGE AFTER OBJECT REMOVAL**

**FIGURE:** row-wise: (a) original image, (b) masked region shows object to be removed, (c) image after object removal
# Table of Contents

1. Outline

2. Markov Feature

3. Localizing Seam Insertions
   - Deterministic Approach
   - Statistical Approach

4. Rotation/Scale Detection

5. Conclusion
A First Attempt at Detecting Seam Carving/Insertions and Localizing Seam Insertions

Outline of the Contributions

(1) We are able to detect whether an image has been subjected to seam carving/insertions (this content-aware image resizing technique was proposed in [1]), for a large enough fraction of seam carving/insertion. We use a machine learning framework using block-based transform domain Markov features, a popular steganalysis feature but unexplored for forensics purposes.
A First Attempt at Detecting Seam Carving/Insertions and Localizing Seam Insertions

Outline of the Contributions (contd.)

(2) We achieve *near-accurate localization of the inserted seams* using prior knowledge of the seam insertion algorithm. We also show the performance of a stochastic approach to seam insertion localization without using prior knowledge.

(3) We also show empirically that *Markov features* are useful for *rotation/scale detection*.
Digital image forensics aims at detecting tampering in digital images.

There are methods to detect tampering due to re-sampling, by assuming that an interpolation kernel (not image specific) was used to scale the entire image.

But, what if the image content is tampered in a “content-aware” manner, i.e. using “seam carving”?

Here, the overall image dimensions are changed but the dimensions of “important regions” are left unchanged. Also, seam carving/insertion can be used to remove objects.
Digital image forensics aims at detecting tampering in digital images.

There are methods to detect tampering due to re-sampling, by assuming that an interpolation kernel (not image specific) was used to scale the entire image.

But, what if the image content is tampered in a “content-aware” manner, i.e. using “seam carving”?

Here, the overall image dimensions are changed but the dimensions of “important regions” are left unchanged. Also, seam carving/insertion can be used to remove objects.
SEAM CARVING VS SCALING AND CROPPING

SEAM CARVING = CONTENT BASED CHANGES

- As said before, seam carving changes image dimensions in an “automatic content-aware” manner.
- Cropping is almost impossible to detect - it can however only remove pixels from the sides or image ends.
- Scaling is also oblivious to the image content and is generally applied to the entire image.

Here, we have considered mainly deletion and insertion of vertical seams.
For a steganalysis problem, the focus is to identify statistical features which change consistently after hiding.

Here, for detecting seam carving, the focus is on identifying common statistical changes even though the seams removed depend on the image content.

Our aim was to see if there are common steganalysis features that can be incorporated for seam carving detection - Markov features worked well to detect seam carving.
**Intuition Behind Using Markov Features**

- **Example of removed seam**
  - Pixels lying to left or right of pixels along the removed seams have their neighborhood changed.
  - Change is in pixel domain neighbors and it also leads to change in local frequency domain statistics.

Choose which ever is more consistent among pixel domain/frequency domain statistics.
Why Does Markov Feature Perform Well

- Markov features based on first order differences in the quantized Discrete Cosine Transform (DCT) domain (the 324-dim feature by Shi et al [4] is called Shi-324) are used.

- Intuition - the pixel neighbors change upon seam removal for the pixels bordering the seams. When many seams are removed, there is a significant change in adjoining pixel statistics as well as in local frequency domain statistics.

- We have experimented with first order difference based 2D histograms in both pixel and frequency (quantized DCT) domains - it is seen that the change in DCT domain is more consistent for proper classification.
Why Does Markov Feature Perform Well

- Markov features based on first order differences in the quantized Discrete Cosine Transform (DCT) domain (the 324-dim feature by Shi et al [4] is called Shi-324) are used.

- **Intuition** - the pixel neighbors change upon seam removal for the pixels bordering the seams. When many seams are removed, there is a significant change in adjoining pixel statistics as well as in local frequency domain statistics.

- We have experimented with first order difference based 2D histograms in both pixel and frequency (quantized DCT) domains - it is seen that the change in DCT domain is more consistent for proper classification.
Markov features based on first order differences in the quantized Discrete Cosine Transform (DCT) domain (the 324-dim feature by Shi et al [4] is called Shi-324) are used.

Intuition - the pixel neighbors change upon seam removal for the pixels bordering the seams. When many seams are removed, there is a significant change in adjoining pixel statistics as well as in local frequency domain statistics.

We have experimented with first order difference based 2D histograms in both pixel and frequency (quantized DCT) domains - it is seen that the change in DCT domain is more consistent for proper classification.
For seam carving, the mathematical relation between two pixels with a seam removed between them is not known.

For seam insertion, there is a linear relationship between the new pixels introduced during seam insertion.

By exploiting the fact that an inserted seam consists of 8-connected pixels, an inserted seam can be recognized.

We also present a statistical approach to localize seam insertions - this uses a probability-map based feature which is high-valued at pixels which are smooth or are linear combinations of their nearby pixels (possible locations of seam insertions).
# Table of Contents

1. Outline

2. **Markov Feature**

3. Localizing Seam Insertions
   - Deterministic Approach
   - Statistical Approach

4. Rotation/Scale Detection

5. Conclusion
The JPEG 2D array (set of $8 \times 8$ quantized DCT magnitudes, computed for every $8 \times 8$ image block) is denoted by $F(u, v)$, $u \in [0, S_u - 1]$ and $v \in [0, S_v - 1]$, where $S_u$ and $S_v$ denote the size of the JPEG 2D array along the horizontal and vertical directions.

The first order difference arrays are expressed as:

- horizontal direction: $F_h(u, v) = F(u, v) - F(u + 1, v)$,
- vertical direction: $F_v(u, v) = F(u, v) - F(u, v + 1)$,
- diagonal direction: $F_d(u, v) = F(u, v) - F(u + 1, v + 1)$,
- minor diagonal direction: $F_m(u, v) = F(u + 1, v) - F(u, v + 1)$
**Laplacian-like Distribution of Difference 2D Arrays**

- Normalized distribution: $p(x)$
- Highly tapered Laplacian-like distribution

First order difference value: $x$
Since the distribution of the elements in the difference 2D arrays is similar to a Laplacian, with a highly peaky nature near 0, the difference values are considered in the range \([-T, T]\). We use \(T=4\).

Each of the difference 2-D arrays is modeled using Markov random process - a transition probability matrix is used to represent the Markov process. The elements in the 4 matrices (one for each direction) are as follows.
**Transition Probability Matrices Explained**

\[
\begin{align*}
    p_h(m, n) &= \frac{\sum_{u,v} \delta(F_h(u, v) = m, F_h(u + 1, v) = n)}{\sum_{u,v} \delta(F_h(u, v) = m)} \\
    p_v(m, n) &= \frac{\sum_{u,v} \delta(F_v(u, v) = m, F_v(u, v + 1) = n)}{\sum_{u,v} \delta(F_v(u, v) = m)} \\
    p_d(m, n) &= \frac{\sum_{u,v} \delta(F_d(u, v) = m, F_d(u + 1, v + 1) = n)}{\sum_{u,v} \delta(F_d(u, v) = m)} \\
    p_m(m, n) &= \frac{\sum_{u,v} \delta(F_m(u + 1, v) = m, F_m(u, v + 1) = n)}{\sum_{u,v} \delta(F_m(u + 1, v) = m)}
\end{align*}
\]

where \( m, n \in \{-T, \cdots, 0, \cdots, T\} \), the summation range for \( u \) is from 0 to \( S_u - 2 \), and for \( v \) from 0 to \( S_v - 2 \), and

\[
\delta(A = m, B = n) = \begin{cases} 
1 & \text{if } A = m \text{ & } B = n \\
0 & \text{otherwise}
\end{cases}
\]
Obtaining a 324-dimensional Feature Vector

Each of the probability matrices \((p_h, p_v, p_d, \text{ and } p_m)\) used to represent the 2-D difference arrays \((F_h, F_v, F_d, \text{ and } F_m)\) respectively have \((2T + 1)^2\) bins.

Therefore, the total feature vector size for Shi-324 is

\[81 \times 4 = 324\] (after converting each probability matrix to an 81-dim vector and concatenating the 4 vectors).
We vary the fraction of seams that is removed from (or inserted to) the image. For example, consider 20% seam carving (20% of the columns in the original image are removed) as our positive examples.

We now divide the entire dataset into an equal number of training and testing images. For each set, we perform seam-carving on half of the images and keep the rest unmodified.

An SVM-based model is learned from the training images. We also investigate the generality of the trained model as a seam-carved test image can have a seam carving percentage different from 20%.
**Table:** Seam-carving detection accuracy for different training-testing combinations: “mixed” refers to that case where the dataset used for training/testing consists of images with varying seam-carving percentages.

<table>
<thead>
<tr>
<th></th>
<th>test 10%</th>
<th>test 20%</th>
<th>test 30%</th>
<th>test 50%</th>
<th>test mixed</th>
</tr>
</thead>
<tbody>
<tr>
<td>train 10%</td>
<td>65.75</td>
<td>66.54</td>
<td>66.26</td>
<td>64.91</td>
<td>70.60</td>
</tr>
<tr>
<td>train 20%</td>
<td>69.11</td>
<td><strong>70.36</strong></td>
<td>70.50</td>
<td>69.11</td>
<td>75.72</td>
</tr>
<tr>
<td>train 30%</td>
<td>74.00</td>
<td>75.54</td>
<td><strong>77.31</strong></td>
<td>77.63</td>
<td><strong>83.88</strong></td>
</tr>
<tr>
<td>train 50%</td>
<td>78.24</td>
<td>80.99</td>
<td>84.67</td>
<td><strong>86.72</strong></td>
<td>91.29</td>
</tr>
<tr>
<td>train mixed</td>
<td>71.77</td>
<td>73.36</td>
<td>74.69</td>
<td>74.59</td>
<td><strong>80.37</strong></td>
</tr>
</tbody>
</table>
# Seam Insertion Detection Results Using Markov Features

Table: Seam insertion detection accuracy for different training-testing combinations:

<table>
<thead>
<tr>
<th>test</th>
<th>train</th>
<th>10%</th>
<th>20%</th>
<th>30%</th>
<th>50%</th>
<th>mixed</th>
</tr>
</thead>
<tbody>
<tr>
<td>10%</td>
<td>68.55</td>
<td>70.47</td>
<td>68.53</td>
<td>63.59</td>
<td>76.95</td>
<td></td>
</tr>
<tr>
<td>20%</td>
<td>76.36</td>
<td><strong>81.88</strong></td>
<td>84.64</td>
<td>81.38</td>
<td><strong>84.63</strong></td>
<td></td>
</tr>
<tr>
<td>30%</td>
<td>80.09</td>
<td>84.65</td>
<td><strong>88.49</strong></td>
<td>93.04</td>
<td><strong>85.71</strong></td>
<td></td>
</tr>
<tr>
<td>50%</td>
<td>82.01</td>
<td>87.41</td>
<td>91.49</td>
<td><strong>95.32</strong></td>
<td><strong>88.84</strong></td>
<td></td>
</tr>
<tr>
<td>mixed</td>
<td>76.74</td>
<td>81.09</td>
<td>83.28</td>
<td>83.34</td>
<td><strong>84.03</strong></td>
<td></td>
</tr>
</tbody>
</table>
Table of Contents

1. Outline
2. Markov Feature
3. Localizing Seam Insertions
   - Deterministic Approach
   - Statistical Approach
4. Rotation/Scale Detection
5. Conclusion
Figure: (a) \( \{a_{i,j}\}_{i=1,j=1}^{i=4,j=5} \) and (b) \( \{b_{i,j}\}_{i=1,j=1}^{i=4,j=6} \) are the image matrices before and after seam insertion, respectively. For points in the first row: 
\[
\begin{align*}
b_{1,1} &= a_{1,1}, & b_{1,2} &= a_{1,2}, & b_{1,3} &= \text{round}\left(\frac{a_{1,2} + a_{1,3}}{2}\right), & b_{1,4} &= \text{round}\left(\frac{a_{1,3} + a_{1,4}}{2}\right), \\
b_{1,5} &= a_{1,4}, & b_{1,6} &= a_{1,5}.
\end{align*}
\]
Consider the image matrix $b$ after seam insertion. For the new pixel values introduced after seam insertion,

\[
\begin{align*}
  b_{i,j} &= \text{round}(\frac{a_{i,j-1} + a_{i,j}}{2}), \\
  b_{i,j+1} &= \text{round}(\frac{a_{i,j} + a_{i,j+1}}{2}).
\end{align*}
\]
Deterministic Approach

UNDERSTANDING LINEAR RELATIONSHIP BETWEEN RESULTANT PIXELS

Consider the image matrix $b$ after seam insertion. For the new pixel values introduced after seam insertion,

\[ b_{i,j} = \text{round}\left(\frac{a_{i,j-1} + a_{i,j}}{2}\right), \]
\[ b_{i,j+1} = \text{round}\left(\frac{a_{i,j} + a_{i,j+1}}{2}\right). \]
Deterministic Approach

Understanding Linear Relationship between Resultant Pixels

Without rounding, and using $a_{i,j+1} = b_{i,j+2}$, and $a_{i,j-1} = b_{i,j-1}$,

$$b_{i,j+1} - b_{i,j} = \frac{(a_{i,j+1} - a_{i,j-1})}{2} = \frac{b_{i,j+2} - b_{i,j-1}}{2},$$

due to rounding, the modified condition is

$$|(2.b_{i,j} - b_{i,j-1}) - (2.b_{i,j+1} - b_{i,j+2})| = 0, \text{ or } 1,$$

we set

$$P_{i,j} = 1 \text{ if } |(2.b_{i,j} - b_{i,j-1}) - (2.b_{i,j+1} - b_{i,j+2})| = 0, \text{ or } 1$$

$$P_{i,j} = 0 \text{ otherwise}$$
Without rounding, and using \( a_{i,j+1} = b_{i,j+2} \), and \( a_{i,j-1} = b_{i,j-1} \),

\[
b_{i,j+1} - b_{i,j} = \frac{(a_{i,j+1} - a_{i,j-1})}{2} = \frac{b_{i,j+2} - b_{i,j-1}}{2},
\]

due to rounding, the modified condition is

\[
|(2.b_{i,j} - b_{i,j-1}) - (2.b_{i,j+1} - b_{i,j+2})| = 0, \text{ or } 1,
\]

we set

\[
P_{i,j} = \begin{cases} 1 & \text{if } |(2.b_{i,j} - b_{i,j-1}) - (2.b_{i,j+1} - b_{i,j+2})| = 0, \text{ or } 1 \\ 0 & \text{otherwise} \end{cases}
\]
Deterministic Approach

Understanding Linear Relationship Between Resultant Pixels

(i) before seam insertion

\[ a_{i,j-2} \quad a_{i,j-1} \quad \text{seam insertion path} \quad a_{i,j} \quad a_{i,j+1} \quad a_{i,j+2} \]

(ii) after seam insertion

\[ b_{i,j-2} \quad b_{i,j-1} \quad b_{i,j} \quad b_{i,j+1} \quad b_{i,j+2} \quad b_{i,j+3} \]

Without rounding, and using \( a_{i,j+1} = b_{i,j+2} \), and \( a_{i,j-1} = b_{i,j-1} \),

\[
\begin{align*}
b_{i,j+1} - b_{i,j} &= \frac{(a_{i,j+1} - a_{i,j-1})}{2} = \frac{b_{i,j+2} - b_{i,j-1}}{2},
\end{align*}
\]

due to rounding, the modified condition is

\[
|2b_{i,j} - b_{i,j-1} - (2b_{i,j+1} - b_{i,j+2})| = 0, \text{ or } 1, \quad (4)
\]

we set

\[
P_{i,j} = \begin{cases} 1 & \text{if } |2b_{i,j} - b_{i,j-1} - (2b_{i,j+1} - b_{i,j+2})| = 0, \text{ or } 1 \\ 0 & \text{otherwise} \end{cases}
\]

\[
\begin{align*}
P_{i,j} &= 1 \\
0 & \text{otherwise}
\end{align*}
\]
Obtaining a Useful Binary Matrix

When the seam passes through the image border, \(\{a_{i,j}, a_{i,j+1}\}\) (\(a_{i,j}\) or \(a_{i,j+1}\) is a seam pixel) gets modified to \(\{b_{i,j}, b_{i,j+1}, b_{i,j+2}\}\), where \(b_{i,j} = a_{i,j}, b_{i,j+1} = \text{round}(\frac{a_{i,j}+a_{i,j+1}}{2})\) and \(b_{i,j+2} = a_{i,j+1}\).

For a pixel one pixel away from a bordering column,

\[
\begin{align*}
    \text{we set } P_{i,j} = 1 & \quad \text{if } (2b_{i,j} - (b_{i,j-1} - b_{i,j+1})) = 0, \text{ or } 1 \\
    & \quad \text{otherwise}
\end{align*}
\]

Once we have this binary matrix \(P\), the next issue is to convert this matrix into an useful signature.
Every seam consists of 8-connected pixels. By tracking points which have a ‘1’-valued parent (except pixels in the first row) and child node (except pixels in the last row) using (7), we obtain the binary matrix $P_B$ from $P$.

$$P_B(i, j) = \begin{cases} 1 & \text{if } P_{i,j} = 1 \\ 0, & \text{otherwise} \end{cases}$$

and

$$\max \{ P_{i-1,j-1}, P_{i-1,j}, P_{i-1,j+1} \} = 1,$$

and

$$\max \{ P_{i+1,j-1}, P_{i+1,j}, P_{i+1,j+1} \} = 1.$$
• **We want that one/more seams detected for seam-inserted images:** the matrix $P_B$ should be non-zero for seam-inserted images. Due to many seam intersections or due to noise (e.g. compression attack), seams may not be detected leading to missed detections. A possible solution is to relax the conditions for finding a seam point.

• **We want that no seams should be detected for original images:** there should not be false detections, i.e. $P_B$ should be an all-zero matrix for a normal image. However, due to presence of smooth regions, even un-tampered images may demonstrate the presence of seams by satisfying (5) (or (6)).
POSSIBLE DETECTION AND ERROR SCENARIOS

- **We want that one/more seams detected for seam-inserted images**: the matrix $P_B$ should be non-zero for seam-inserted images. Due to many seam intersections or due to noise (e.g. compression attack), seams may not be detected leading to missed detections. A possible solution is to relax the conditions for finding a seam point.

- **We want that no seams should be detected for original images**: there should not be false detections, i.e. $P_B$ should be an all-zero matrix for a normal image. However, due to presence of smooth regions, even un-tampered images may demonstrate the presence of seams by satisfying (5) (or (6)).
Deterministic Approach

**IDENTIFYING SEAM LOCATIONS**

**FIGURE:**

row-wise: (a) after 3% seam insertion, (b) $P$ matrix, (c)/(d) points with valid parent/child nodes, (e) $P_B$ matrix
**Deterministic Approach**

**Identifying Seam Locations**

**Figure:**
row-wise: (a) after 3% seam insertion, (b) $P$ matrix, (c)/(d) points with valid parent/child nodes, (e) $P_B$ matrix
**Examples of Seam Insertion Localization**

**Figure:**
row-wise: (a)/(b)/(c) after 1%/5%/10% seam insertion, (d)/(e)/(f) the detected seams are shown for (a)/(b)/(c), respectively
Deterministic Approach

Examples of Seam Insertion Localization

Figure:
row-wise: (a)/(b)/(c) after 1%/5%/10% seam insertion, (d)/(e)/(f) the detected seams are shown for (a)/(b)/(c), respectively
EXAMPLES OF SEAM INSERTION LOCALIZATION

Figure:
row-wise: (a)/(b)/(c) after 1%/5%/10% seam insertion, (d)/(e)/(f) the detected seams are shown for (a)/(b)/(c), respectively
Examples of Seam Insertion Localization

**Figure:**
row-wise: (a)/(b)/(c) after 1%/5%/10% seam insertion, (d)/(e)/(f) the detected seams are shown for (a)/(b)/(c), respectively
We can discard columns pertaining to smooth regions while computing $P$ and $P_B$.

For this task, we obtain the Canny edge-map [2] and discard columns for which the spacing between two consecutive edge-points along a certain column is quite high (or when the column has no edge point).

E.g. let there be $m$ edge points for the $j^{th}$ column in a $N_1 \times N_2$ image, denoted by $\{e_{i,j}\}_{i=1}^{m}$. The successive difference values are $\{d_{i,j}\}_{i=1}^{m-1}$, where $d_{i,j} = e_{i+1,j} - e_{i,j}$. If the maximum separation between successive edge points exceeds a threshold, i.e. $\max_i d_{i,j} > s_{frac} \cdot N_1$, we remove the $j^{th}$ column, where $s_{frac}$ is a tunable parameter.
We can discard columns pertaining to smooth regions while computing $P$ and $P_B$.

For this task, we obtain the Canny edge-map [2] and discard columns for which the spacing between two consecutive edge-points along a certain column is quite high (or when the column has no edge point).

E.g. let there be $m$ edge points for the $j^{th}$ column in a $N_1 \times N_2$ image, denoted by $\{e_{i,j}\}_{i=1}^{m}$. The successive difference values are $\{d_{i,j}\}_{i=1}^{m-1}$, where $d_{i,j} = e_{i+1,j} - e_{i,j}$. If the maximum separation between successive edge points exceeds a threshold, i.e. $\max_i d_{i,j} > s_{frac} \cdot N_1$, we remove the $j^{th}$ column, where $s_{frac}$ is a tunable parameter.
We can discard columns pertaining to smooth regions while computing $P$ and $P_B$.

For this task, we obtain the Canny edge-map [2] and discard columns for which the spacing between two consecutive edge-points along a certain column is quite high (or when the column has no edge point).

E.g. let there be $m$ edge points for the $j^{th}$ column in a $N_1 \times N_2$ image, denoted by $\{e_{i,j}\}_{i=1}^m$. The successive difference values are $\{d_{i,j}\}_{i=1}^{m-1}$, where $d_{i,j} = e_{i+1,j} - e_{i,j}$. If the maximum separation between successive edge points exceeds a threshold, i.e. $\max_i d_{i,j} > s_{frac} \cdot N_1$, we remove the $j^{th}$ column, where $s_{frac}$ is a tunable parameter.
**Deterministic Approach**

**Using Smoothness Constraint**

**Figure:**

(a) 3% seam-inserted image, (b) & (c) $P_B$ for seam-inserted & original image, (d) smoothed seam-inserted image ($s_{frac} = 0.98$), (e) & (f) $P_B$ for smoothed seam-inserted & original image.
Deterministic Approach

**Using Smoothness Constraint**

**Figure:**

(a) 3% seam-inserted image, (b) & (c) $P_B$ for seam-inserted & original image, (d) smoothed seam-inserted image ($s_{frac} = 0.98$), (e) & (f) $P_B$ for smoothed seam-inserted & original image
For noisy images, the exact conditions for seam insertion detection may not be met and hence, they need to be “relaxed”. The “relaxed conditions” to compute $P$ are:

$$
P_{i,j} = \begin{cases} 
1 & \text{if } |(2b_{i,j} - b_{i,j-1}) - (2b_{i,j+1} - b_{i,j+2})| \leq \delta_1 \\
0 & \text{otherwise}
\end{cases}$$

where $\delta_1$ is increased to allow more pixels to be labeled as ‘1’. For a pixel one pixel away from a bordering column,

$$
P_{i,j} = \begin{cases} 
1 & \text{if } |(b_{i,j} - \frac{(b_{i,j-1} - b_{i,j+1})}{2})| \leq \delta_2 \\
0 & \text{otherwise}
\end{cases}$$

where the number of ‘1’s increases by increasing $\delta_2$. For JPEG $QF=100$, we use $\delta_1 = 3$ (was 1) and $\delta_2 = 1.5$ (was 0.5).
For noisy images, the exact conditions for seam insertion detection may not be met and hence, they need to be “relaxed”. The “relaxed conditions” to compute $P$ are:

$$P_{i,j} = \begin{cases} 1 & \text{if } |(2b_{i,j} - b_{i,j-1}) - (2b_{i,j+1} - b_{i,j+2})| \leq \delta_1 \\ 0 & \text{otherwise} \end{cases}$$

(8)

where $\delta_1$ is increased to allow more pixels to be labeled as ‘1’.

For a pixel one pixel away from a bordering column,

$$P_{i,j} = \begin{cases} 1 & \text{if } |(b_{i,j} - \frac{(b_{i,j-1} - b_{i,j+1})}{2})| \leq \delta_2 \\ 0 & \text{otherwise} \end{cases}$$

(9)

where the number of ‘1’s increases by increasing $\delta_2$. For JPEG $QF=100$, we use $\delta_1 = 3$ (was 1) and $\delta_2 = 1.5$ (was 0.5).
For noisy images, the exact conditions for seam insertion detection may not be met and hence, they need to be “relaxed”. The “relaxed conditions” to compute $P$ are:

$$P_{i,j} = \begin{cases} 1 & \text{if } |(2b_{i,j} - b_{i,j-1}) - (2b_{i,j+1} - b_{i,j+2})| \leq \delta_1 \\ 0 & \text{otherwise} \end{cases}$$

(8)

where $\delta_1$ is increased to allow more pixels to be labeled as ‘1’. For a pixel one pixel away from a bordering column,

$$P_{i,j} = \begin{cases} 1 & \text{if } |(b_{i,j} - \frac{(b_{i,j-1} - b_{i,j+1})}{2})| \leq \delta_2 \\ 0 & \text{otherwise} \end{cases}$$

(9)

where the number of ‘1’s increases by increasing $\delta_2$. For JPEG $QF=100$, we use $\delta_1 = 3$ (was 1) and $\delta_2 = 1.5$ (was 0.5).
Deterministic Approach

Does Using Relaxed Conditions Help?

**Figure:**
row-wise: (a)/(b) False alarm ($P_{FA}$) and missed detection ($P_{MD}$) rates are computed for JPEG images/ JPEG images with “relaxed conditions”.

**Table:**

<table>
<thead>
<tr>
<th>smoothness factor $s_{frac}$</th>
<th>PFA</th>
<th>PMD (40%)</th>
<th>PMD (30%)</th>
<th>PMD (20%)</th>
<th>PMD (10%)</th>
<th>PMD (5%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

PFA and PMD values for JPEG QF = 100.
Deterministic Approach

How Does Detection Accuracy Vary?

Figure:
row-wise: (a)/(b) detection error rates are computed for JPEG images/ JPEG images with "relaxed conditions."
To simultaneously estimate a pixel’s probability of being a linear combination of its neighboring pixels and the weights of the combination, an Expectation Maximization (EM) algorithm [3] is used.

The output is a probability matrix (probability-map or p-map) where a high value indicates higher probability that the pixel has been re-sampled.

During seam insertion, the seam pixel is removed and then replaced by two pixels whose values are the average of the seam pixel’s left and right neighbors. This is similar to re-sampling and the pixels that are inserted are correlated with its neighbors.
Using EM to Localize Seam Insertions

- From the p-map obtained using EM on a seam-inserted image, the higher values correspond to possible seam insertion locations.
- We threshold the p-map to obtain a binary matrix $P$, where the cutoff value is optimally chosen based on a number of training images (half of which contain seam insertions) to maximize the detection accuracy.
- Then, we find 8-connected paths in $P$ to obtain locations of possible seams.

For 10% seam insertion, for the best choice of threshold, the accuracy (to detect seam inserted images) is 63%.
# Table of Contents

1. Outline

2. Markov Feature

3. Localizing Seam Insertions
   - Deterministic Approach
   - Statistical Approach

4. **Rotation/Scale Detection**

5. Conclusion
**Scale Detection**

**Table:** Detection of scaling using Shi-324 feature and SVM models for different train-test combinations:

<table>
<thead>
<tr>
<th></th>
<th>test 0.25</th>
<th>test 0.50</th>
<th>test 0.75</th>
<th>test 0.95</th>
<th>test 1.05</th>
<th>test 1.25</th>
<th>test 1.50</th>
<th>test 2.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>train 0.25</td>
<td><strong>95.20</strong></td>
<td>92.08</td>
<td>80.89</td>
<td>40.31</td>
<td>39.05</td>
<td>49.11</td>
<td>49.67</td>
<td>48.70</td>
</tr>
<tr>
<td>train 0.50</td>
<td>87.93</td>
<td><strong>92.50</strong></td>
<td><strong>87.47</strong></td>
<td>38.58</td>
<td>37.05</td>
<td>46.69</td>
<td>48.09</td>
<td>46.97</td>
</tr>
<tr>
<td>train 0.75</td>
<td>63.56</td>
<td>68.45</td>
<td><strong>73.53</strong></td>
<td>44.08</td>
<td>44.92</td>
<td>46.69</td>
<td>48.60</td>
<td>47.76</td>
</tr>
<tr>
<td>train 0.95</td>
<td>49.81</td>
<td>48.88</td>
<td>51.16</td>
<td><strong>60.81</strong></td>
<td>56.10</td>
<td>49.72</td>
<td>50.37</td>
<td>50.28</td>
</tr>
<tr>
<td>train 1.05</td>
<td>38.77</td>
<td>40.63</td>
<td>46.41</td>
<td>65.24</td>
<td><strong>74.98</strong></td>
<td>60.07</td>
<td>56.48</td>
<td>55.92</td>
</tr>
<tr>
<td>train 1.25</td>
<td>47.02</td>
<td>44.69</td>
<td>39.42</td>
<td>60.02</td>
<td>66.36</td>
<td><strong>88.96</strong></td>
<td>69.25</td>
<td>60.90</td>
</tr>
<tr>
<td>train 1.50</td>
<td>31.83</td>
<td>26.37</td>
<td>21.11</td>
<td>77.68</td>
<td><strong>86.11</strong></td>
<td><strong>91.52</strong></td>
<td><strong>96.41</strong></td>
<td><strong>88.49</strong></td>
</tr>
<tr>
<td>train 2.00</td>
<td>26.93</td>
<td>21.71</td>
<td>16.17</td>
<td>81.92</td>
<td><strong>90.35</strong></td>
<td><strong>93.10</strong></td>
<td><strong>98.70</strong></td>
<td><strong>98.23</strong></td>
</tr>
</tbody>
</table>
**Rotation Detection**

**Table:** Rotation detection: (0.6,10) means that the image is first rotated by 10° and then 60% of the image is retained per dimension.

<table>
<thead>
<tr>
<th>test</th>
<th>train</th>
<th>0.6,10</th>
<th>0.6,20</th>
<th>0.6,30</th>
<th>0.6,40</th>
<th>0.8,10</th>
<th>0.8,20</th>
<th>0.8,30</th>
<th>0.8,40</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.6,10</td>
<td>73.67</td>
<td>71.48</td>
<td>59.65</td>
<td>58.71</td>
<td>72.74</td>
<td>66.64</td>
<td>57.08</td>
<td>54.33</td>
<td></td>
</tr>
<tr>
<td>0.6,20</td>
<td>68.31</td>
<td>88.12</td>
<td>87.56</td>
<td>87.05</td>
<td>67.94</td>
<td>83.50</td>
<td>75.49</td>
<td>65.10</td>
<td></td>
</tr>
<tr>
<td>0.6,30</td>
<td>64.82</td>
<td>89.93</td>
<td>94.04</td>
<td>94.55</td>
<td>64.77</td>
<td>87.98</td>
<td>86.53</td>
<td>75.68</td>
<td></td>
</tr>
<tr>
<td>0.6,40</td>
<td>63.89</td>
<td>90.31</td>
<td>94.45</td>
<td>95.39</td>
<td>63.65</td>
<td>89.14</td>
<td>88.21</td>
<td>79.36</td>
<td></td>
</tr>
<tr>
<td>0.8,10</td>
<td>73.21</td>
<td>71.81</td>
<td>59.37</td>
<td>58.90</td>
<td>72.93</td>
<td>69.71</td>
<td>58.53</td>
<td>55.87</td>
<td></td>
</tr>
<tr>
<td>0.8,20</td>
<td>69.52</td>
<td>87.88</td>
<td>87.14</td>
<td>87.47</td>
<td>70.08</td>
<td>89.05</td>
<td>87.56</td>
<td>78.19</td>
<td></td>
</tr>
<tr>
<td>0.8,30</td>
<td>65.94</td>
<td>89.19</td>
<td>93.43</td>
<td>93.94</td>
<td>67.05</td>
<td>94.04</td>
<td>95.71</td>
<td>94.69</td>
<td></td>
</tr>
<tr>
<td>0.8,40</td>
<td>64.12</td>
<td>89.61</td>
<td>94.64</td>
<td>95.29</td>
<td>66.08</td>
<td>95.25</td>
<td>96.51</td>
<td>96.09</td>
<td></td>
</tr>
</tbody>
</table>
If training is performed using 20°, 30° or 40°, the detection accuracy is high if the rotation angle for the test images is equal to or higher than the rotation angle for training images.

If the size of the test images is higher than the size of the training images, the detection accuracy is higher.
GENERALIZABILITY OF SHI-324 FEATURE

- For a practical setting, a variety of SVM models, based on different rotation and scale factors, can be used to detect whether an image is rotated or scaled.

- We have shown that the Shi-324 feature is generalizable for a variety of tamper operations - seam carving/insertion, rotation and scaling.
TABLE OF CONTENTS

1 Outline

2 Markov Feature

3 Localizing Seam Insertions
   • Deterministic Approach
   • Statistical Approach

4 Rotation/Scale Detection

5 CONCLUSION
We have presented a machine learning based approach where DCT domain Markov features are shown to be useful for detecting seam carving and seam insertions.

We have proposed an algorithm which exploits the linear relationship between pixels located on/near the seam to detect seam insertions. Assuming knowledge of the seam insertion method, we obtain highly accurate localization of the inserted seams.
Future work shall involve making the seam insertion framework more general so that the newly introduced pixels can be any arbitrary combination of neighboring pixels.

The machine learning based approach needs to be further improved upon to increase the detection rate when the seam carving/insertion percentage is low enough.

We also intend to localize the seam carving region by using a block-based approach and multi-class decision based on the Markov features.
S. Avidan and A. Shamir.
Seam carving for content-aware image resizing. 

J. Canny.
A computational approach to edge detection. 

A. C. Popescu and H. Farid.
Exposing digital forgeries by detecting traces of re-sampling. 

A Markov process based approach to effective attacking JPEG steganography.