

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/282150847>

# Tamper Detection of JPEG Image Due to Seam Modifications

ARTICLE *in* IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY · DECEMBER 2015

Impact Factor: 2.41 · DOI: 10.1109/TIFS.2015.2464776

---

READS

34

6 AUTHORS, INCLUDING:



Kanoksak Wattanachote  
National Central University

13 PUBLICATIONS 7 CITATIONS

[SEE PROFILE](#)



Timothy K. Shih  
National Central University

332 PUBLICATIONS 1,458 CITATIONS

[SEE PROFILE](#)



Hon-Hang Chang  
National Central University

26 PUBLICATIONS 628 CITATIONS

[SEE PROFILE](#)

# Tamper Detection of JPEG Image Due to Seam Modifications

Kanoksak Wattanachote, Timothy K. Shih, *Senior Member, IEEE*, Wen-Lung Chang and Hon-Hang Chang

**Abstract**—Content-aware image retargeting has been investigated since the last decade as a paradigm of image modification for proper display on the different screen sizes. Modifications, such as seam carving or seam insertion, have been introduced to achieve aforesaid image retargeting. The changes in an image are not easily recognizable by human eyes. Inspired by the Blocking Artifact Characteristics Matrix (BACM), a method to detect tampers caused by seam modification on JPEG retargeted images without knowledge of the original image is proposed in this paper. In a BACM block matrix, we found that the original JPEG image demonstrates a regular symmetrical data, whereas the symmetrical data in a block reconstructed by seam modification is destroyed. Twenty-two features are proposed to train the data by using a Support Vector Machine (SVM) classification method. The experimental results clearly demonstrate that the proposed method provides a very high recognition rate for those JPEG retargeted images. The source codes and the complete experimental data can be accessed at <http://video.minelab.tw/DETS/>.

**Index Terms**—Image forensics, JPEG analysis, Seam-carving detection, Steganalysis features, Tamper detection

## I. INTRODUCTION

NOWADAYS, tampering digital images without leaving of obvious traces becomes easier with the rapid growth of image processing techniques. The recognition of counterfeiting digital image is one of the most useful techniques in relevant areas. One of the counterfeiting digital image methods is content-aware image retargeting technique. The main idea of retargeting techniques is to protect important

Accepted by IEEE Transaction on Image Forensics and Security on July 20, 2015.

Kanoksak Wattanachote is with the Department of Computer Science and Information Engineering, National Central University, No. 300, Jhongda Road, Jhongli City, Taoyuan County 32001, Taiwan (R.O.C.) (e-mail: kanoksak.wattanachote@gmail.com).

Timothy K. Shih (corresponding author) is with the Department of Computer Science and Information Engineering, National Central University, No. 300, Jhongda Road, Jhongli City, Taoyuan County 32001, Taiwan (R.O.C.) (e-mail: timothykshih@gmail.com).

Wen-Lung Chang was with the Department of Computer Science and Information Engineering, National Central University, No. 300, Jhongda Road, Jhongli City, Taoyuan County 32001, Taiwan (R.O.C.) (e-mail: tobethappy@gmail.com).

Hon-Hang Chang is with the Department of Computer Science and Information Engineering, National Central University, No. 300, Jhongda Road, Jhongli City, Taoyuan County 32001, Taiwan (R.O.C.) (e-mail: sicachang@gmail.com).

content in the image that can be recognized as high energy pixels. Two variations of seam modifications, namely seam carving and seam insertion, are used for image retargeting to reduce or enlarge the size of an image properly to the target sizes by gracefully carving out or inserting pixels at different locations. Once the seam carving or seam insertion method was applied on an image, the image is a tampered.

Multimedia forensics is a multiple-disciplinary research field with important impacts on protection of public safety and enhancement of security. In multimedia forensics, steganography detection and forgery detection are active areas and are continuing to face challenges. For instance, JPEG-based image forgery and detection, as well as the relevant manipulations including double JPEG compression, image rescaling, copying-pasting, inpainting, and compositing, have been studied [32, 33, 34, 35, 36]. While most image forensics methods target on traditional image tampering, seam-modification-based image tampering in JPEG format has been ignored to some extent. In this research, we consider the problem of image forensics for tampered JPEG image due to seam modifications with no knowledge of the original image.

Over the past few years, several tamper detection techniques for image have been studied and researched. For instance, the tampered image detection technique for resampling image has been proposed in [1-3]. Both upscaling and downscaling technique are useful for resample artificial detection. Gordon *et al.* [8] proposed the Expectation Maximization (EM) algorithm to solve the problem for detection purpose. Gallagher [1] proposed an algorithm operated by exploiting the property that the second derivative signal of the interpolated images contains a periodicity. Kirchner [9] proposed the interpolation detection algorithm to exploit a periodicity in the second derivative signal and then to compute the Discrete Fourier Transform (DFT) of interpolated images to determine the interpolation factor for detection. Popescu *et al.* [2] showed how to find the pattern automatically in any region of an image by defining the form using the correlations in images and finally use the EM algorithm to detect the image forgery by resampling. Popescu *et al.* [3] showed that there is a simpler method to detect the resampling forgery based on the variance of the second order difference of image pixels. Farid [10] arranged multiple splices in the existing methods about image forgery and detection. As a whole, the image forgery and detection can be classified into: pixel-based, cloning, resampling, splicing,

statistical, and format-based methods. Cloning is one of the most common image forgery methods used to clone a part of image to another image. Cloning detection method is similar to seam insertion detection, but the method for cloning detection is not suitable for seam carving detection. In general, kernel based interpolation methods for resampling detection usually fail when the resizing is done by seam modification to maintain the important features. Then, the unimportant area would be resized or removed based on the seam modification method used. In fact, the continuity for the consecutive pixels is destroyed. Recently, Liu and Chen [31] propose a calibrated neighboring-joint density-based approach, with a simple feature set to distinguish steganograms and tampered images from untouched images. The accuracy for seam-tampered detection is significantly high. However, the tampered image in their experiment is obtained by seam modifications in order to preserve the image size after removing an object, rather than protecting the important content.

One of the important techniques relevant to JPEG compression was studied and implemented for tamper detection [13, 21]. The Discrete Cosine Transform (DCT) block-based compression approach is an important process, which leads the distinguished characteristic of JPEG [17, 18, 25, 26, 27, 28, 29, 30]. Fridrich *et al.* [11] proposed the DCT block method to detect cloning images by lexicographically sorting the DCT block coefficients and grouping similar blocks with the same spatial offset in the image. Sarkar *et al.* [13] used the Markov features depending on first order differences in the quantized DCT domain. By combining Markov features and SVM. Sarkar *et al.* [13] proposed the seam-tampered detection method, which is useful to detect seam-tampered images. In addition, Wei *et al.* [21] proposed a method using  $2 \times 2$  blocks as mini-squares to search for one of the nine surrounding patches by using Cosine Similarity for seam carving detection, and to test the Uncompressed Color Image Database (UCID) that consists of 1338 TIFF images, which are the images also used in [6, 13, 21]. Popescu *et al.* [12] proposed another approach by implementing the Principal Component Analysis (PCA) on small fixed size image blocks.

In this paper, we adopted a machine learning technique for two-class classification between the seam-tampered JPEG image and the original JPEG image. In the experiment, we found that the relationship between pixels in a seam-tampered JPEG image would always be changed. Therefore, the main idea for detection is to identify common statistical changes even though the seam locations are depended on the content of image. In theory, BACM is a matrix that expresses the symmetric blocking effect in a JPEG image [4]. It is usually applied to find feature vectors for the seam-carving image detection. Even if few seams were removed, the continuity in an image can be destroyed, especially due to the blocking effect. We found that the feature extraction from BACM can efficiently detect seam-tampered images. We use an SVM-based model [5] to train features extracted from BACM. We also demonstrated the effectiveness of our detection method on UCID database [6]. We compared our accuracy results to the accuracy results derived from Sarkar *et al.*'s [13]

method and Wei *et al.*'s [21] method, and discussed their detection methods.

Aforementioned, we surveyed many counterfeiting digital image detection methods and found that only a few could effectively detect the tampered image generated by seam modifications. This is due to the fact that removal or insertion of pixels in seam-modifying images could not be detected and restored with retrospective modification by any backward formulas. In addition, recognition of tampered digital image is possible by finding localization of tampering on image as in [13, 20, 23, 24]. Our methods in recognition of tampered image can find the location of seam insertion. We can detect the tampering of seam insertion in tampered JPEG as demonstrated in the Discussion. However, since the original seams after deletion is lost, we are not able to find seam carving locations in the tampered image without their originals. In this paper, we propose a tamper detection method of JPEG image caused by seam modifications, whereas the localization of seam modifications is one of our future work. Hence, to evaluate the performance of our framework, we demonstrate our results derived from different factors to show that our method gains much higher accuracy than the existing techniques.

The rest of this paper is organized as follows. Section II devotes to related work and theoretical concepts. Section III describes our proposed method, including the BACM scenario and calculations. Experimental results and analysis are demonstrated in Section IV, and discussed in Section V. Finally, conclusions and suggestions for future work are drawn in Section VI.

## II. RELATED WORK

### A. Seam Modification

Seam modifications in this paper explicitly refer to both seam carving and seam insertion. Seam carving (or seam removal) is an operation to reduce the size of an image by gracefully carving out a set of connected pixels discovered by an energy function which is based on the energy of pixels along a certain path. The definition of a seam was described as a set of connected pixels that traverses the image in the vertical (from top to bottom) or horizontal (from left to right) layout. Avidan and Shamir [7] proposed the optimal seam  $s^*$  that minimizes the seam cost by (1). With an energy function  $e$ , we can define the cost of a seam using (2), and use the seam carving method using the energy function in (3).

$$s^* = \arg \min_s E(s) \quad (1)$$

$$E(s) = \sum_{i=1}^n e(I(s_i)) \quad (2)$$

$$e(I) = \left| \frac{\partial}{\partial x} I \right| + \left| \frac{\partial}{\partial y} I \right| \quad (3)$$

The optimal seam can be found using a dynamic programming algorithm. In vertical approach, the first step is to traverse the image from the second row to the last row. Then the cumulative minimum energy  $M$  in (4), is computed for all possible connected seams for each entry  $(i, j)$  as,

$$M(i, j) = e(i, j) + \operatorname{argmin}(M(i-1, j-1), M(i-1, j), M(i-1, j+1)), \quad (4)$$

where  $e$  is an energy of the current pixel at  $(i, j)$  along the height and width of the image respectively. The last step is to traverse back the minimum value in the last row from  $M$  to connect the vertical seam as shown in Fig. 2(b)-(c).

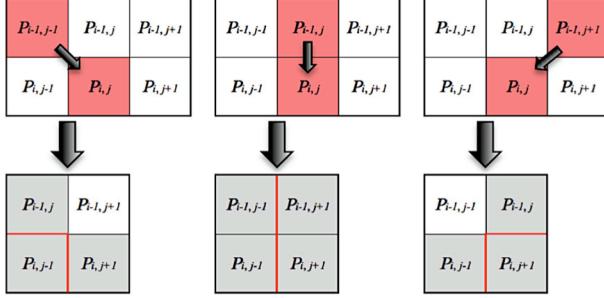


Fig. 1. The calculation schemes of the three possible vertical seam step costs for pixel  $P_{i,j}$  using forward energy. After removing the seam (red), new neighbors (gray) are created. (a) Seam removal from upper left to center. (b) Seam removal from the middle. (c) Seam removal from upper right to center.

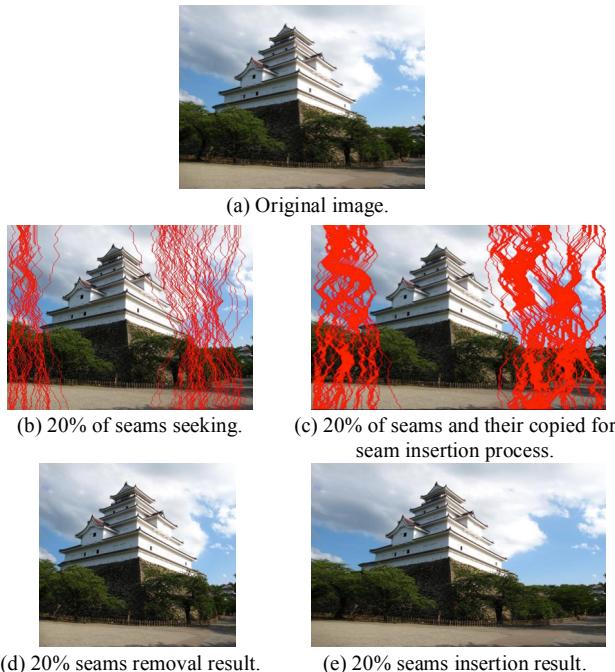


Fig. 2. The demonstration of seam modifications. (a) The original image. (b) The image with 20% vertical seams before removal. (c) The image with 20% vertical seams and their copies in red marks. (d) The image of 20% vertical seams removal. (e) The image of 20% vertical seams insertion.

For seam removal, we first calculate the energy function in the original image, Fig. 2(a), and find the optimal seam to delete. After a seam was removed, the energy function should be recalculated, and the new optimal seam is reiteratively discovered for the next seam removal. We repeat these stages until the targeted size of image is reached. An example is shown in Fig. 2(d). Rubinstein *et al.* [16] proposed the forward energy method. It can accurately find the optimal seams for object removal by inserting another energy functions based on the new edges created by previously non-adjacent pixels that become neighbors once the seam was removed. The real change in energy after removal of a seam can be measured as shown in Fig. 1. Rubinstein *et al.* [16], defined costs

respectively for each of the three possible cases as,

$$C_L(i, j) = |I(i, j+1) - I(i, j-1)| + |I(i-1, j) - I(i, j-1)|, \quad (5)$$

$$C_U(i, j) = |I(i, j+1) - I(i, j-1)|, \quad (6)$$

$$C_R(i, j) = |I(i, j+1) - I(i, j-1)| + |I(i-1, j) - I(i, j+1)|, \quad (7)$$

where  $C_L$ ,  $C_U$  and  $C_R$  in (5), (6), and (7) mean the three possible situations after seam deletion. Then, they use these costs in a new accumulative cost matrix  $M$  to find the seams by dynamic programming. For vertical seams, each cost  $M(i, j)$  is updated by following the rule in (8).  $P(i, j)$  is an original measure of energy, such as the result of high level tasks (e.g., edge detection) or user supplied weight, which is used on the top of the forward energy cost. This method proposed an effective content aware resizing method for images.

$$M(i, j) = P(i, j) + \operatorname{argmin} \begin{cases} M(i-1, j-1) + C_L(i, j) \\ M(i-1, j) + C_U(i, j) \\ M(i-1, j+1) + C_R(i, j) \end{cases}. \quad (8)$$

Seam insertion can be thought as the reverse of seam removal [22]. Suppose  $n$  seams are to be inserted, the aforesaid processes for seam insertion must be performed first in order to find those  $n$  seams. The coordinates and sequence of those  $n$  seams are recorded. Finally, the coordinates are used to insert new seams to the original image in sequence. For example, first we have to copy all the optimal seams as founded and shown in Fig. 2(b). Then we replace the seam pixel  $a_{i,j}$  by two pixels  $b_{i,j}$  and  $b_{i,j+1}$ , whereas  $b_{i,j}$  is represented as,

$$b_{i,j} = \operatorname{round}\left(\frac{a_{i,j-1} + a_{i,j}}{2}\right), \quad (9)$$

and the result is shown in Fig. 2(e).

### B. Seam Tamper Detection in Image Dataset

Several seam modification techniques and tamper detection methods have been proposed. Lu *et al.* [20] proposed the forensic hash technique by determining the stable SIFT points in the image. The image is transmitted and undergoes seam carving, geometric transforms and tampering operations by keeping those SIFT points. The proposed forensic hash construction can be extended to accurately estimate seam carving and to detect local tampering. The estimation accuracy was evaluated by the probability of correct detection and probability of false detection.

For seam-tampered detection on modified image, Sarkar *et al.* [13] used Markov features, which are calculated by the first order differences in the quantized DCT domain. The domain is subsequently referred to as Shi-324 [14], by defining four matrixes, and transforming the matrices to Markov features (Shi-324). Finally, by combining Markov features and SVM, the work in [13] proposed a strong seam carving detection method.

Aforementioned, the idea is to find the Shi-324 feature. The first step is to transform the image to DCT domain for all  $8 \times 8$  blocks. The next step is to define four arrays in four directions: horizontal, vertical, diagonal and minor diagonal. These arrays are defined as,

$$F_h(u, v) = F(u, v) - F(u+1, v), \quad (10)$$

$$F_v(u, v) = F(u, v) - F(u, v+1), \quad (11)$$

$$F_d(u, v) = F(u, v) - F(u+1, v+1), \quad (12)$$

$$F_m(u, v) = F(u+1, v) - F(u, v+1), \quad (13)$$

where  $F(u, v)$  denotes a given image,  $u \in [0, S_u - 1]$ , and  $v \in [0, S_v - 1]$ .  $S_u$  and  $S_v$  denote width and height of the image, respectively. The elements in those 2D arrays are similar to the Laplacian operators. The different values in the array are considered to be in the range  $[-T, T]$ . Here, if the values in the array are larger than  $T$  or less than  $-T$ , we map those values to  $[-T, T]$ . Sarkar *et al.* [13] set  $T = 4$ . So, the number of possible value in each array is  $(2T + 1)^2 = 81$ . Then, we model each different 2D array by using Markov random process. It is a transition probability matrix to present the Markov process. The probability matrices are,

$$P_h(m, n) = \frac{\sum_{u,v} \delta(F_h(u, v) = m, F_h(u+1, v) = n)}{\sum_{u,v} \delta(F_h(u, v) = m)}, \quad (14)$$

$$P_v(m, n) = \frac{\sum_{u,v} \delta(F_v(u, v) = m, F_v(u, v+1) = n)}{\sum_{u,v} \delta(F_v(u, v) = m)}, \quad (15)$$

$$P_d(m, n) = \frac{\sum_{u,v} \delta(F_d(u, v) = m, F_d(u+1, v+1) = n)}{\sum_{u,v} \delta(F_d(u, v) = m)}, \quad (16)$$

$$P_m(m, n) = \frac{\sum_{u,v} \delta(F_m(u+1, v) = m, F_m(u, v+1) = n)}{\sum_{u,v} \delta(F_m(u, v) = m)}. \quad (17)$$

The probability matrices  $P_h$ ,  $P_v$ ,  $P_d$ , and  $P_m$  present the 2D difference arrays such as  $F_h$ ,  $F_v$ ,  $F_d$ , and  $F_m$  respectively. We have  $(2T + 1)^2 = 81$  features in each array. Hence, the total vector size is  $81 \times 4 = 324$  in Shi-324 features, where  $m, n \in [-T, T]$ ; and, the summation ranges for  $u$  and  $v$  are from 0 to  $S_u - 2$  and from 0 to  $S_v - 2$ , respectively. The function of  $\delta(m, n)$  is defined as,

$$\delta(m = A, n = B) = \begin{cases} 1, & \text{if } m = A \text{ and } n = B \\ 0, & \text{otherwise} \end{cases}, \quad (18)$$

where  $A, B \in [-H, H]$ , and  $H = 4$ .

### III. THE PROPOSED METHOD

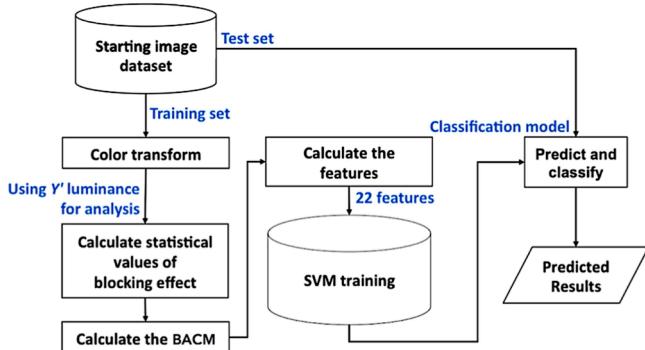


Fig. 3. The system architecture diagram for seam-tampered detection of the image dataset.

This research introduces a seam-tampered detection method for the JPEG retargeted images. We first describe the blocking effect of JPEG images. After that, we explain the BACM to

measure the symmetrical property of blocking artifacts introduced by JPEG encoder. Moreover, we propose twenty-two features, including other eight features especially for seam-tampered detection obtained from BACM. At the end, we construct the hyper-plane by using SVM classifier to train the feature vectors which were derived from the BACM in operation before and after seam carving. The workflow of system is shown in Fig. 3. To increase readability of the paper, we create a table of notations in Table I.

TABLE I  
TABLE OF NOTATIONS

Notation	Definition
UCID	Uncompressed Color Image Database, contains the TIFF images that have been used in [13] and [21] (1338 images).
QF	The Quality Factor of JPEG compression.
UCUS	“Unknown Compression / Undefined Source and Size” image database, contains JPEG images (1009 images) at unknown QF from several sources in different sizes.
TIFF	An uncompressed image type (.tif).
JPEG	A compressed image type (.jpg).
QF75	The compressed image at QF value equal to 75.
DCT	The Discrete Cosine Transform (DCT) block-based compression is the transformation process that leads the distinguished characteristic of JPEG.
BACM	Blocking Artifact Characteristics Matrix that contains the symmetrical property of blocking artifacts introduced by JPEG encoder.
SVM	Support Vector Machine classification method.
‘A’-‘H’	A through H are the values of the pixels in the positions depicted in Fig. 4.
Z'(i, j)	Absolute different value of pixels in block calculated by  A-B-C+D .
Z''(i, j)	Absolute different value of pixels in block calculated by  E-F-G+H .
H <sub>i</sub> (n)	The normalized histogram of Z'(i, j).
H <sub>ii</sub> (n)	The normalized histogram of Z''(i, j).
K(i, j)	Absolute different value of normalized histograms Z'(i, j) and Z''(i, j).
B(i, j)	The average of K(i, j), as in (22).

#### A. System Architecture

##### 1) Starting image dataset

The starting images are derived from images in the UCID database [6] and the UCUS database. These images consist of different types, such as landscape, character, static artificial items, and artwork. The UCID images are in the .tif format (uncompressed image). The UCUS images derived from our collection are in .jpg format (compressed image) with unknown or undefined quality factor (QF). After we obtained the images, we transform all images to JPEG compressed format by down sampling to a 4:2:2 YCbCr color space with a defined QF at 75 (QF75), in accordance with Sarkar *et al.*’s approach. The image dataset consists of two parts; one consists of the images in their original size, and another consists of the images in the retargeting size with a defined QF compression after retargeting. The dataset is prepared to use for training set and test set. The detail of the dataset setup is described in the experiments.

##### 2) Color space

We transformed the data into the YCbCr color space. After the down-sampling step for JPEG compressed format, we found that the luminance,  $Y'$ , in YCbCr is not commonly

affected. This is the reason for us to choose  $Y'$  in our analysis.

### 3) Calculate statistical values of blocking effect

The statistical calculation of the blocking effect is presented in section B. In this step, the main idea is to calculate the different values and the statistics of the intensity distribution for each block. The distinction of characteristic in each block was generated during the JPEG compression processes, is described in section B. The block characteristic was generated based on the DCT transformation and the Entropy coding processes.

### 4) Calculate the BACM

We calculated the matrix to show the symmetry phenomena in blocking effects. The detail about the BACM symmetry phenomena is given in section C. At this stage, the goal is to calculate twenty-two feature vectors from the BACM by identifying the symmetry of the matrix. The detailed method is described in section D.

### 5) SVM training

We trained the twenty-two features and constructed the classification model or training model to identify the retargeted image in the dataset.

### 6) Identification

In the last stage, the training model is used to test the test set in a cross validation approach.

## B. Detection of JPEG Blocking Effects

In JPEG compression [17, 18, 25, 26, 27, 28, 29, 30], after the down-sampling process, the image is divided into small  $8 \times 8$  non-overlapped blocks. We transformed each block to the DCT domain and divided each block by quantized table. Then entropy coding is performed to yield a data stream in the final stage. The JPEG compression is one of the lossy compression methods. Therefore, we look for blocking signatures in the images as an evidence of JPEG compression.

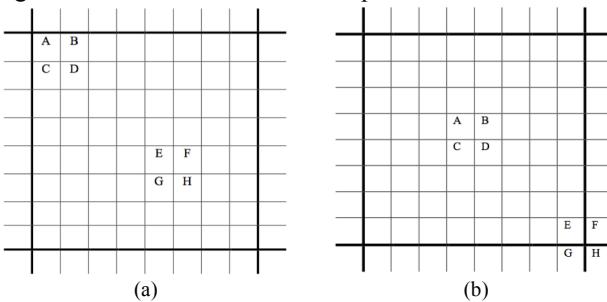


Fig. 4. Demonstration of the relative position of A through H pixels in  $8 \times 8$  blocks, where A through H are the values of the pixels in those positions. (a) The position of A is at (1, 1). (b) The position of A is at (4, 4), in the center of the block.

When the compression process of JPEG image is performed, even with a light compression, it may result in small pieces of discontinuities consistency in block boundaries. The main idea proposed in [4] and [15] shows that the pixels' difference across blocks boundary should be similar to those within blocks without compression. As shown in Fig. 4, assuming a block grid is known and every single grid means a pixel in the image, we can calculate the difference in every single grid within the blocks and across a

block boundary. The center of block results the highest significant difference. For each block in every  $(i, j)$ , we compute the  $Z'(i, j)$  and  $Z''(i, j)$  which are the differences between the neighbor pixels. The  $Z'(i, j)$  and  $Z''(i, j)$  are defined as,

$$Z'(i, j) = |A - B - C + D|, \quad (19)$$

$$Z''(i, j) = |E - F - G + H|, \quad (20)$$

where A to H are the values of pixels at the position shown in Fig. 4(a). The entry  $(i, j)$  is denoted as the location of A in each block. The position of B to H in each block is changed according to the position of A. After calculating the difference among adjacent pixels, we calculate the normalized histograms  $H_I(n)$  and  $H_{II}(n)$  of  $Z'(i, j)$  and  $Z''(i, j)$ , respectively, where  $H_I(n)$  means the intensity of  $Z'(i, j)$  in  $(i, j)$  for each block and n is the number of blocks. Fig. 5(b) and (c) show the histograms of blocking effect of uncompressed and compressed image, respectively, derived from the image in Fig. 5(a). Fig. 5(d) shows the difference of blocking effect between uncompressed and compressed images. The blue line means  $Z'(i, j)$  and green line means  $Z''(i, j)$  at position A = (4, 4). In the diagram, the histogram of uncompressed image for  $H_I(n)$  and  $H_{II}(n)$  are similar. The histogram of compressed image has significant differences between  $H_I(n)$  and  $H_{II}(n)$ . Then, the differences of K between  $H_I(n)$  and  $H_{II}(n)$  at  $(i, j)$  are calculated as,

$$K_{(i,j)}(n) = |H_I(n) - H_{II}(n)|. \quad (21)$$

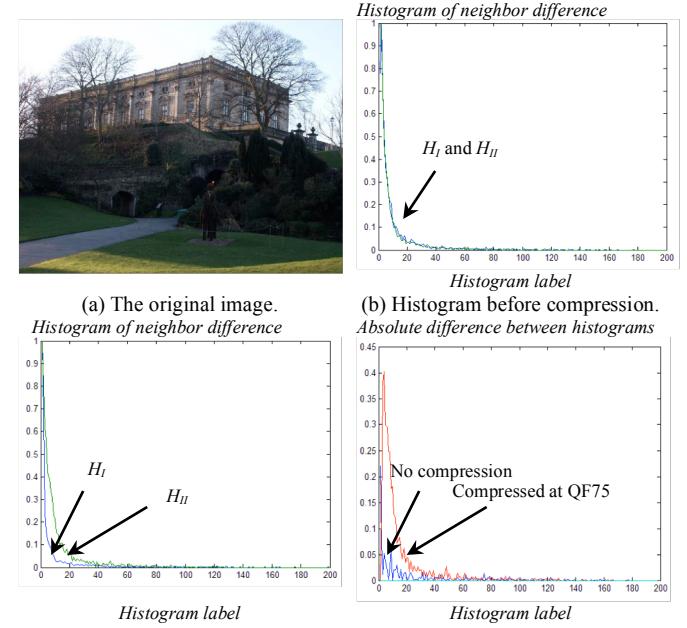


Fig. 5. Demonstration of the histogram of neighbor differences comparing region  $H_I$  and  $H_{II}$ . (a) The original image from UCID database. (b) No compression is applied to the image. (c) The image compressed at QF75. (d) Difference in histograms ( $H_I$  and  $H_{II}$ ) for both cases.

## C. The Symmetry Phenomena of Blocking Effects

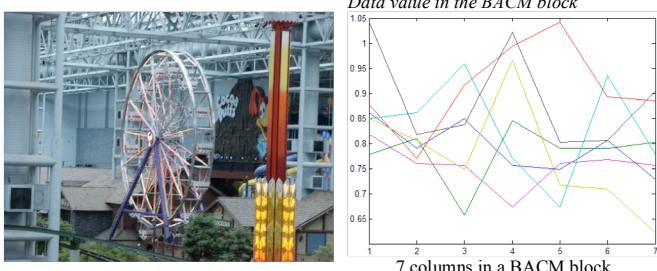
The different position of parameters A through H in Fig. 4, in accordance with the example data in a block of compressed image in Fig. 6, can cause the  $K_{(i,j)}(n)$  to have different values at different  $(i, j)$ . The result of blocking pattern in Fig. 4(b)

shows that if the position of  $A$  lies in the middle column of block, where  $A = (4, y)$ , for each row  $y \in [1, 7]$ , each block will have a significant difference. After the value at the center of each row in the block has been calculated, we found that the value at the middle is the largest and will gradually decrease along the both sides. Moreover, the value of  $K$  between  $A = \{(4, y), (4, 8-y)\}$ ,  $y \in [1, 3]$  varies in a small degree. The symmetry phenomenon for each row is drawn as in Fig. 7(c). This phenomenon does occur not only for the rows but also in the columns. In [15], they used the BACM for the cropped JPEG image detection. First, we divided an image into small non-overlapped  $8 \times 8$  blocks. For each block  $(i, j)$ , we calculate  $K_{(i,j)}(n)$  and get the average  $K$ , which is denoted as  $B(i, j)$  below,

$$B(i, j) = \frac{\sum K_{(i,j)}(n)}{255 * 2 + 1} \quad (22)$$

0.8767	0.7006	0.5988	2.1957	0.5440	0.6810	0.9941	2.6810
0.7006	0.5636	0.7241	2.2935	0.7515	0.7319	0.6262	2.2935
0.7476	0.6536	0.8141	2.4344	0.9159	0.7319	0.8063	2.2466
2.0861	1.8630	2.2857	3.5969	2.1879	1.7926	2.1566	0.9667
0.5284	0.6928	0.9863	2.6732	0.8611	0.6575	0.5832	2.2153
0.7945	0.7006	0.6301	2.2935	0.6693	0.5714	0.7123	2.3718
0.9276	0.7045	0.8180	2.2348	0.7358	0.6419	0.8102	2.4736
2.2622	1.8239	2.2114	0.9315	2.0744	1.8787	2.3405	3.6008

Fig. 6. An example block of the BACM calculation for a compressed image.



(a) The original image.

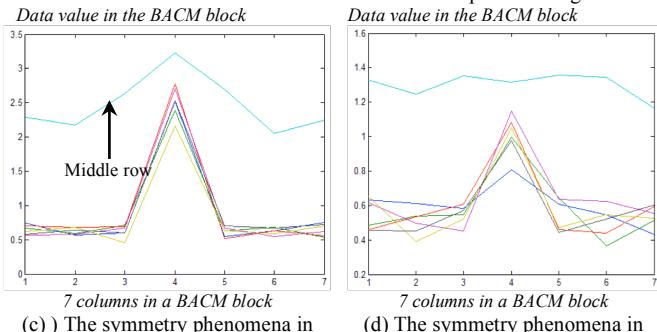


Fig. 7. Demonstration of the comparison of symmetry phenomena in uncompressed, compressed and retargeted image in BACM. (a) The original image. (b) The symmetry in uncompressed image. (c) The symmetry in compressed image. (d) The symmetry in retargeted JPEG image.

Fig. 6 shows an example of BACM where the symmetry phenomena in BACM can be found in a DCT matrix block. The red square is the main symmetry phenomena region. From the matrix, we can see that the data is symmetrical to the middle, diagonal, vertical, and horizontal directions. Here, we can find the regular rule from the matrix and obtain the training vector.

Fig. 7 shows the symmetry phenomena of BACM in uncompressed, JPEG (compressed), and retargeted image. In

Fig. 7(b)-(d), the seven different color lines represent the values of each row in the BACM block. Here, the Cyan line represents the values of the middle row in BACM that has the most significant different values in the matrix. In Fig. 7(b), there is no symmetry phenomenon in uncompressed image. In Fig. 7(c), the symmetry phenomena could be found in a JPEG (compressed) image. We observed that the values  $B(i, j)$  of the matrix  $B$  in JPEG (compressed) image demonstrated in (23), become regular. In Fig. 7(d), the retargeting image resized by seam carving was experimented by BACM. And, the result shows that the symmetry phenomena in the retargeted image were destroyed.

#### D. The BACM Feature Vectors

Sarkar *et al.* [13] presented fourteen feature vectors derived from BACM that have been used to detect the modified images. We implement those fourteen features, including our new additional features, specifically for seam tamper detection in all image dataset.

$$B = \begin{bmatrix} B_{(1,1)} & B_{(1,2)} & B_{(1,3)} & B_{(1,4)} & B_{(1,5)} & B_{(1,6)} & B_{(1,7)} & B_{(1,8)} \\ B_{(2,1)} & B_{(2,2)} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ B_{(3,1)} & \cdot \\ B_{(4,1)} & \cdot & \cdot & B_{(4,4)} & \cdot & \cdot & \cdot & \cdot \\ B_{(5,1)} & \cdot \\ B_{(6,1)} & \cdot & \cdot & \cdot & \cdot & \cdot & B_{(6,7)} & \cdot \\ B_{(7,1)} & \cdot & \cdot & \cdot & \cdot & B_{(7,6)} & B_{(7,7)} & \cdot \\ B_{(8,1)} & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & B_{(8,8)} \end{bmatrix} \quad (23)$$

In the first calculation, we divide the  $8 \times 8$  block matrix  $B$ , into seven non-overlapped parts, including R1, R2, R3, and R4. The horizontal parts are denoted by H. And, the vertical parts are denoted by V. Finally, the center point  $B(4, 4)$  is denoted by C, as shown in Fig. 8. The regional distribution is defined in equations (24)-(30).

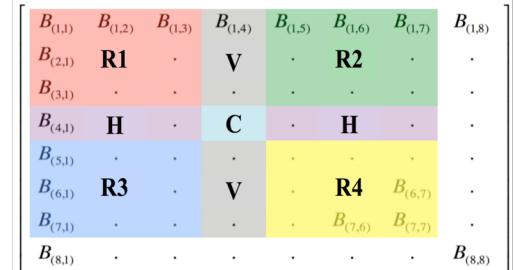


Fig. 8. Demonstration of regional distribution mapping in  $8 \times 8$  block of the BACM.

$$R1 : \{B(i, j) | 1 \leq i, j < 4\} \quad (24)$$

$$R2 : \{B(i, j) | 1 \leq i < 4; 4 < j < 8\} \quad (25)$$

$$R3 : \{B(i, j) | 4 < i < 8; 1 \leq j < 4\} \quad (26)$$

$$R4 : \{B(i, j) | 4 < i, j < 8\} \quad (27)$$

$$H : \{B(4, j) | 1 \leq j < 4; 4 < j < 8\} \quad (28)$$

$$V : \{B(i, 4) | 1 \leq i < 4; 4 < i < 8\} \quad (29)$$

$$C : \{B(4, 4)\} \quad (30)$$

Then, we construct fourteen features [13] for seam-modified image detection. The first two features are the symmetry in the region H (28) and V (29) around the point in

the center C (30), described in (31)-(32). The next six features, describe the symmetry of the four regions R1, R2, R3, and R4 around H, V, and C, as demonstrated in (33)-(38).

$$F_1 = \sum_{j=1}^3 |B(4,j) - B(4,8-j)| \quad (31)$$

$$F_2 = \sum_{i=1}^3 |B(i,4) - B(8-i,4)| \quad (32)$$

$$F_3 = \sum_{i=1}^3 \sum_{j=1}^3 |B(i,j) - B(i,8-j)| \quad (33)$$

$$F_4 = \sum_{i=5}^7 \sum_{j=1}^3 |B(i,j) - B(i,8-j)| \quad (34)$$

$$F_5 = \sum_{i=1}^3 \sum_{j=1}^3 |B(i,j) - B(8-i,j)| \quad (35)$$

$$F_6 = \sum_{i=1}^3 \sum_{j=5}^7 |B(i,j) - B(8-i,j)| \quad (36)$$

$$F_7 = \sum_{i=1}^3 \sum_{j=1}^3 |B(i,j) - B(8-i,8-j)| \quad (37)$$

$$F_8 = \sum_{i=1}^3 \sum_{j=1}^3 |B(i,8-j) - B(8-i,j)| \quad (38)$$

After that, we calculate the probability at center point C occupied in the region R1, R2, R3, R4, V, and H, respectively.

$$F_9 = C / \sum_{i=1}^3 \sum_{j=1}^3 B(i,j) \quad (39)$$

$$F_{10} = C / \sum_{i=1}^3 \sum_{j=5}^7 B(i,j) \quad (40)$$

$$F_{11} = C / \sum_{i=5}^7 \sum_{j=1}^3 B(i,j) \quad (41)$$

$$F_{12} = C / \sum_{i=5}^7 \sum_{j=5}^7 B(i,j) \quad (42)$$

$$F_{13} = C / \sum_{j=1}^7 B(4,j) - C \quad (43)$$

$$F_{14} = C / \sum_{i=1}^7 B(i,4) - C \quad (44)$$

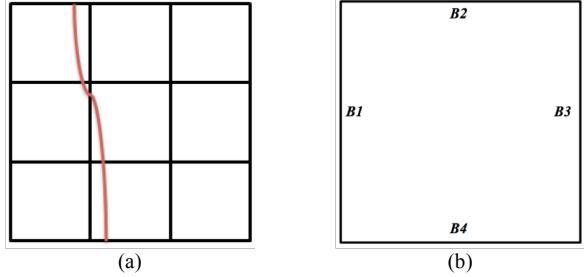


Fig. 9. (a) Demonstration of the seam that can make the fault and discontinuity for the boundary in JPEG image. (b) An expression of four boundaries in BACM.

As an important contribution, we proposed new eight features for seam tamper image detection. In the  $8 \times 8$  BACM, we found the difference in four directions. First, we define four arrays for calculating the difference between pixels, denoted as  $B_h(i,j)$ ,  $B_v(i,j)$ ,  $B_d(i,j)$ , and  $B_m(i,j)$ . The differences of arrays are expressed as,

$$\text{Horizontal : } B_h(i,j) = |B(i,j) - B(i+1,j)|, \quad (45)$$

$$\text{Vertical : } B_v(i,j) = |B(i,j) - B(i,j+1)|, \quad (46)$$

$$\text{Diagonal : } B_d(i,j) = |B(i,j) - B(i+1,j+1)|, \quad (47)$$

$$\text{Minor Diagonal : } B_m(i,j) = |B(i+1,j) - B(i,j+1)|, \quad (48)$$

where  $B(i,j)$  denotes the elements of BACM block in JPEG image and  $i,j \in [1,7]$ . Each block consists of four directions of arrays: horizontal ( $h$ ), vertical ( $v$ ), main diagonal ( $d$ ), and minor diagonal ( $m$ ). We calculate the average of difference in arrays to create the four features by,

$$F_k = \sum_{i=1}^7 \sum_{j=1}^7 B_x(i,j) / 49 \quad (49)$$

where  $k$  indicates the features calculated by  $B_x(i,j)$  and  $k=15, 16, 17, 18$  when  $x$  represents  $h, v, d$ , and  $m$ , respectively.

The last four features are the breaking of block boundaries. The deletion and insertion of seam modifications make the boundary faulted and discontinued as shown in Fig. 9. In this case, the four features can be obtained from the difference between boundaries. Assumed the left column, top row, right column and bottom row in BACM are denoted as B1, B2, B3, and B4, respectively. Thus, our proposed four features describe the absolute difference between (B1, B2), (B2, B3), (B3, B4), and (B4, B1). The formulae are,

$$F_{19} = \sum_{i=1}^7 |B(i,1) - B(1,8-i)|, \quad (50)$$

$$F_{20} = \sum_{i=1}^7 |B(i,1) - B(7,i)|, \quad (51)$$

$$F_{21} = \sum_{i=1}^7 |B(8-i,7) - B(7,i)|, \quad (52)$$

$$F_{22} = \sum_{i=1}^7 |B(8-i,7) - B(1,8-i)|, \quad (53)$$

where  $F_{19}$  -  $F_{22}$  denote our last four features and  $B$  is BACM. We have to take the absolute value to avoid the negative value that affects the feature vector.

#### IV. EXPERIMENTAL RESULTS

We conduct detailed experiments for tamper detection analysis by obtaining the images from two databases, namely the UCID database and the UCUS database. The uncompressed images are derived from the UCID database that was used in [13] and [21]. The images in the UCID database are in  $512 \times 384$  pixels. The UCUS database is obtained by collecting the JPEG images at unknown QF from different sources and in various sizes. These images are subsequently compressed in  $512 \times 384$  pixels at QF75 as the starting images. We also conduct an experiment by combining the images from both data sources (totally with 2347 images) for tamper detection to test the independence of images from different sources.

We vary the fraction of seams modification into different tampering rates: 1%, 2%, 5%, 10%, 20%, 30%, and 50%. For example, suppose the image size is  $512 \times 384$  pixels, 10% vertical seam reduction rate means the width size was reduced by 10% resulting the image size becomes  $461 \times 384$  pixels. We divide the entire images into an equal number of training sets and test sets. Hence, we totally have seven sets for training,





The accuracy results are subsequently observed from the last process in Fig. 3, and recorded to demonstrate in Table IV-XI. Fig. 10 demonstrates the average accuracy of cross validation obtained using both the UCID and the UCUS databases, from Table II to Table XI. The cross validation sets are conducted in 8 different tampering rates consist of 1%, 2%, 5%, 10%, 20%, 30%, 50%, and mixed rates; and 5 different tampering rates consist of 10%, 20%, 30%, 50%, and mixed rates. The two charts in Fig. 10 demonstrate that the average accuracy of those two cross validation sets has no significant difference. The highest average accuracy for seam carving detection was found in the experiment with QF100, whereas the lowest was found in the experiment with QF50. Besides, the highest average accuracy for seam insertion detection was found in the experiment with QF10, whereas the lowest was found in the experiment with QF75. In general, we conclude that different QFs can affect tamper detection accuracy.

### C. The Detection Accuracy from Different Approaches

TABLE XII

TEMPER DETECTION RESULT FOR SEAM REDUCTION AT DIFFERENT TAMPER RATES, AT QF 100 FROM DIFFERENT METHODS (a - e).

Test \ Train	10%	20%	30%	50%	Mixed
(a) Sarkar <i>et al.</i> 's proposed method (MM270K database) [13]					
10%	65.75	66.54	66.26	64.91	70.60
20%	69.11	70.36	70.50	69.11	75.72
30%	74.00	75.54	77.31	77.63	83.88
50%	78.24	80.99	84.67	86.72	91.29
Mixed	71.77	73.36	74.69	74.59	80.37
(b) Wei <i>et al.</i> 's proposed method (UCID database) [21]					
10%	64.87	60.15	62.15	70.45	64.40
20%	64.76	92.21	93.11	95.21	86.32
30%	65.28	84.42	92.61	95.28	84.40
50%	61.54	73.08	74.53	95.84	76.25
Mixed	71.36	83.78	94.82	95.96	86.48
(c) Our proposed method (UCID database)					
10%	97.42	93.24	86.10	68.24	94.66
20%	98.39	98.80	95.37	84.72	98.02
30%	98.92	99.10	99.25	94.62	99.18
50%	99.03	99.29	99.44	99.63	99.36
Mixed	98.54	97.42	94.84	86.06	98.77
(d) Our proposed method (UCUS database)					
10%	98.81	94.85	88.40	70.86	97.18
20%	99.31	99.60	97.52	89.05	99.01
30%	99.85	99.85	100	97.42	99.90
50%	98.86	99.55	99.90	100	100
Mixed	99.36	98.51	96.43	89.10	99.70
(e) Our proposed method (UCID & UCUS database)					
10%	97.85	94.91	90.05	71.33	96.49
20%	98.70	98.98	97.08	87.88	98.66
30%	99.17	99.38	99.49	96.19	99.38
50%	99.30	99.55	99.68	99.79	99.55
Mixed	98.66	97.91	95.40	87.84	98.83

According to [13], Sarkar *et al.* proposed their method by the experimental steps to prepare the starting images that can be clearly described in this section. They conducted the experiment with images derived from the MM270K database, whose images are in JPEG format at QF75. In the next step, the JPEG images are decompressed before seam modification so as to create positive training examples. Finally, the images are compressed at QF100. Those images are defined to be the starting image dataset. They also have experimented with uncompressed images from the UCID database. Since the Shi-324 feature works in the quantized DCT domain, the input

image has to be JPEG. Therefore, the UCID images are also subsequently JPEG compressed at QF100. The experimental results in [13], describe that “the detection results of the UCID images are similar to that obtained using JPEG images (MM270K database) as the starting images”. In accordance with our results in Table II to Table XI, the average accuracy obtained from two databases has no significant difference. The accuracy results obtained using the MM270K database in [13] is presented in Table XII (a). Moreover, Wei *et al.* [21], proposed their method and conducted the experiment based on Sarkar *et al.*'s experimental steps for starting images obtained using UCID database. Their accuracy results are demonstrated in [21] (Table 5), which is presented again in our experiment as shown in Table XII (b). Therefore, we conducted the experiments in section A and B based on the steps described in [13], for the images obtained using UCID and UCUS databases. The results demonstrated in section A for the compressed images at QF100 are presents in Table XII (c) and (d). We also conduct an experiment using the combination sets of images, by obtaining the images from both UCID and UCUS database to be starting images (2347 images), so as to validate the bias of different image database. The result is demonstrated in Table XII (e).

The results from the existing and our method are shown in Table XII. The results in Table XII demonstrated in cross validation tests of 5 different tampering rates, in accordance with Sarkar *et al.*'s approach. The mixed dataset was included with the aim to detect the tampered images without knowing seam tampering rates. Table XII (a) shows Sarkar *et al.*'s results and (b) demonstrates Wei *et al.*'s results. Table XII (c)-(e) demonstrates the results derived from our proposed method obtained using our two databases. The results show that the detection accuracy values obtained by our proposed method are higher than those derived from Sarkar *et al.*'s method. In average, the value by our proposed method is also higher than those derived from Wei *et al.*'s method. The experiments in Table XII (c)-(e) aimed to validate the dependence of the detection accuracy and the image databases. The results show that the detection accuracy and the image databases are independent since the results in Table XII (e) is similar to the results in Table XII (c) and (d). The detection accuracy by our proposed method in average is around 98-99%, higher than that obtained by Sarkar *et al.*'s and Wei *et al.*'s methods.

TABLE XIII  
TEMPER DETECTION RESULT FOR SEAM INSERTION AT DIFFERENT TAMPER RATES, AT QF 100 FROM DIFFERENT METHODS (a - b).

Test \ Train	10%	20%	30%	50%	Mixed
(a) Sarkar <i>et al.</i> 's proposed method (MM270K database) [13]					
10%	68.55	70.47	68.53	63.59	76.95
20%	76.36	81.88	84.64	81.38	84.63
30%	80.09	84.65	88.49	93.04	85.71
50%	82.01	87.41	91.49	95.32	88.84
Mixed	76.74	81.09	83.28	83.34	84.03
(b) Our proposed method (UCID database)					
10%	95.89	91.74	85.87	73.73	94.92
20%	98.21	98.39	94.84	87.74	97.83
30%	98.88	98.80	99.33	95.37	98.77
50%	98.80	98.95	99.14	99.40	98.84
Mixed	92.90	87.22	82.40	76.38	94.43

In addition, the experimental results in Table XIII show that the average accuracy of seam insertion detection obtained by our proposed method is also higher than that obtained by Sarkar *et al.*'s method. That means the symmetric phenomenon in blocking effects is evident for the compressed image. Both seam carving and seam insertion can destroy symmetric phenomenon.

#### D. Tamper Detection of Uncompressed Image (TIFF)

We also experiment with uncompressed images obtained using the images from UCID database. The UCID images are retargeted and subsequently passed through the detection processes as described in Fig. 3, without compression process. The experimental results in Table XIV show that test by mixed model can lead the detection accuracy in averages around 67-69%, both in seam reduction and seam insertion detection.

TABLE XIV  
TAMPER DETECTION RESULT FOR SEAM MODIFICATION, AT DIFFERENT TAMPER RATE (UCID UNCOMPRESSED IMAGES).

Test	Train	10%	20%	30%	50%	Mixed
<b>(a) Seam carving</b>						
10%		63.00	55.19	54.67	49.63	54.19
20%		58.52	65.06	64.05	56.95	63.45
30%		53.70	70.96	72.87	66.18	72.35
50%		38.15	76.94	82.81	89.54	85.91
Mixed		52.69	67.00	68.83	65.88	69.47
<b>(b) Seam insertion</b>						
10%		59.71	58.18	58.18	55.64	58.07
20%		65.17	65.17	64.91	63.23	65.51
30%		68.27	69.47	69.96	66.97	69.28
50%		72.91	74.89	66.97	76.35	75.97
Mixed		65.84	66.48	64.50	64.50	66.97

#### E. Tamper Detection of Compressed Images at Unknown QF

TABLE XV  
TAMPER DETECTION RESULT FOR SEAM MODIFICATION AT, UNKNOWN QF (UCID IMAGES).

Test	Train	1%	2%	5%	10%	20%	30%	50%	Mixed
<b>Seam Carving</b>									
1%		84.23	80.01	80.87	80.19	77.99	63.98	50.97	79.86
2%		80.64	83.71	80.79	79.71	77.58	64.80	50.97	79.97
5%		80.87	80.49	84.75	81.43	79.33	68.05	53.62	81.17
10%		79.60	79.41	80.87	85.46	79.93	72.35	56.43	80.46
20%		76.64	76.31	77.84	78.44	85.43	77.17	62.52	79.60
30%		73.39	73.39	74.85	75.41	79.71	87.74	70.48	78.33
50%		66.85	66.89	68.98	70.33	75.67	82.88	92.41	78.96
Mixed		78.03	77.99	79.41	79.56	80.08	74.44	62.78	84.68
<b>Seam Insertion</b>									
1%		83.93	81.54	80.08	79.37	76.91	75.07	73.51	79.30
2%		81.20	83.63	80.83	79.67	77.84	75.78	74.03	79.97
5%		82.03	82.70	85.46	81.84	80.12	77.65	75.93	81.54
10%		84.01	84.42	84.64	87.37	82.44	80.68	78.81	84.30
20%		86.25	86.17	86.36	86.36	90.73	85.87	84.08	87.82
30%		88.49	88.71	89.01	88.98	90.62	92.60	88.71	90.21
50%		92.04	91.12	92.19	91.63	93.72	93.83	96.64	93.50
Mixed		85.43	85.58	85.39	85.01	84.63	82.62	81.09	88.04

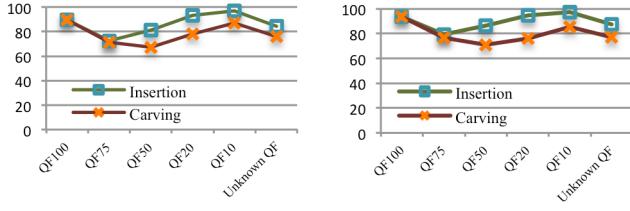


Fig. 11. Demonstrate the average accuracy of tamper detection in different QF data obtained using UCID database.

We conducted an experiment for tampered images with unknown QF, by equally combining the compressed images at different QF for each tampering rate dataset. The tampered images with unknown QF for each dataset are obtained using UCID database. The results are shown in Table XV. Both seam carving and insertion charts in Fig. 11, demonstrate that the average accuracy is slightly higher than that obtained using QF50 but slightly lower than that obtained using QF20 and QF10.

Fig. 11 demonstrates the average accuracy of cross validation obtained using UCID databases. The accuracy results obtained using UCID images are demonstrated in Table II, IV, VI, VIII, X and XV (with unknown QF). The datasets are conducted in 8 different and 5 different tampering rates. Two charts in Fig. 11 demonstrate that the average accuracy of those two cross validation sets has no significant difference. The highest average accuracy for seam carving detection was found in QF100 test, whereas the lowest was found in QF50 test. Besides, the highest average accuracy for seam insertion detection was found in QF10 test, whereas the lowest was found in QF75 test. The detection of tampered images at unknown QF implicitly means an average accuracy obtained from QF10, QF20, QF50, QF75, and QF100 images.

We also conduct a small experiment to find the relationship between seam tamper rates and different energy functions. The experiment collected 500 images from UCID dataset. The detection result shows that no significant difference between using forward and backward energy function for seam tampering. Reasonably, since the symmetry phenomena in BACM will always has high opportunity to be destroyed in accordance with the tamper rate.

## V. DISCUSSION

### A. Seam Tamper Detection

The results in Table II-XI illustrate the model constructed by the mixed training sets is with the highest average accuracy. This means that, the model suitable for test should be more decentralized for all tampering rate datasets, instead of a particular dataset. For example, the hyper-plane built by 50% training set roughly classified the low tampering rate test sets with the lowest average accuracy (only around 50% for seam carving detection and 50-65% for seam insertion detection). However, for the hyper-plane built by 50% training set, the highest average accuracy occurred at the 50% test set, which is around 97%. At QF100, both seam carving and seam insertion detection have the similar accuracy for each training and test set. For the other QF, seam insertion detection accuracy is higher than seam carving detection.

According to the results in Table II-XI, the quality factor can affect the detection accuracy as described in Fig. 10. Our proposed method shows that the highest average accuracy for insertion detection is found in the detection of compressed images at QF10, whereas the highest average accuracy for carving detection is found in the detection of compressed image at QF100. In accordance with Sarkar *et al.*'s and Wei *et al.*'s experiments, they conducted the experiment for seam

carving detection by emphasizing on the compressed images at QF100. The results showed that the average accuracy obtained by our proposed method is higher than that obtained by Sarkar *et al.*'s and Wei *et al.*'s methods, as shown in Table XII. In addition, for uncompressed image dataset results shown in Table XIV, the average accuracy is significantly less than the JPEG compressed image since the symmetric phenomenon in blocking effects is not evident for uncompressed image. However, the starting images have an innate rule that cannot simulate artificially. Therefore, we can still use block to detect the forgery images that are damaged by seam modification methods.

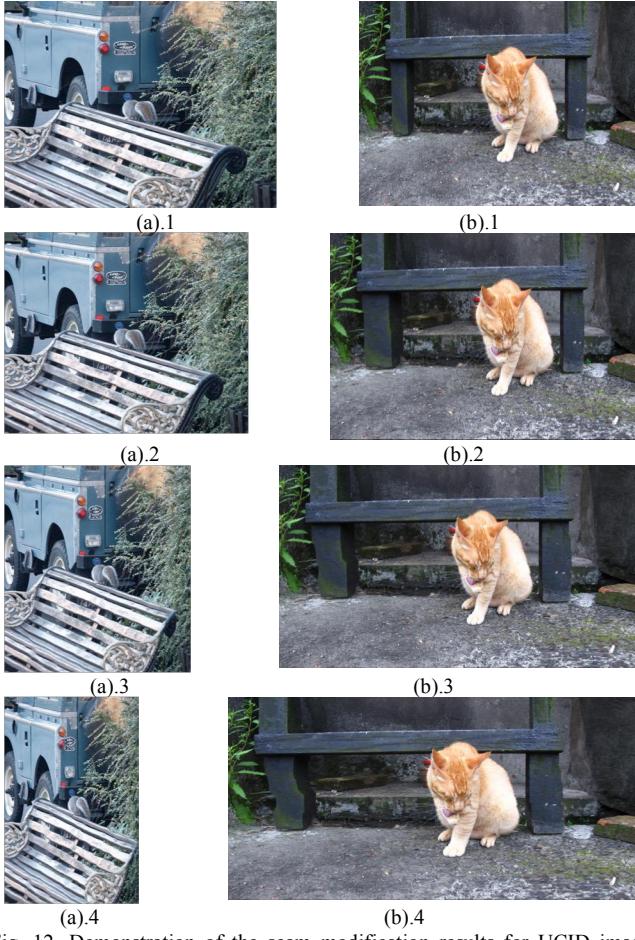


Fig. 12. Demonstration of the seam modification results for UCID image (left) and UCUS image (right). The images from top to bottom in column (a) ((a).1–4) on the left hand side represent original image, and 10%, 30%, and 50% seam carving, whereas the images from top to bottom in column (b) ((b).1–4) on the right hand side represent original image, and 10%, 30%, and 50% seam insertion, respectively.

In our experiments, we have seam modification rates as, 50%, 30%, 20%, 10%, 5%, 2%, and 1%, including the experiment with different QFs as, QF100, QF75, QF50, QF20, and QF10. We conducted an experiment for tamper images with unknown QF, as demonstrated in Table XV, by equally mixing the compressed images with different QF as a dataset for each tampering rate. The detection accuracy for tampered images with unknown QF demonstrates an average accuracy of the detection obtained from QF10, QF20, QF50, QF75, and QF100, as shown in Fig. 11 (a) and (b). We observed that the

quality of image in different QF (as shown in Fig. 13) can affect the accuracy of tamper detection.

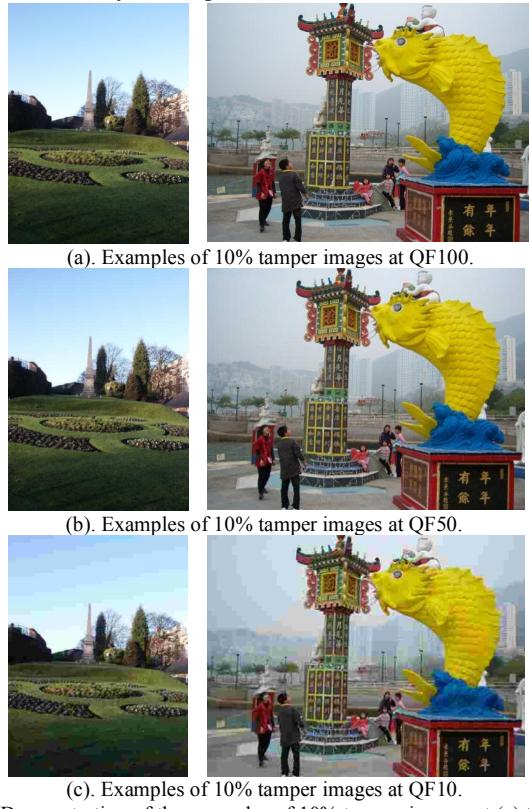


Fig. 13. Demonstration of the examples of 10% tamper images at (a) QF100, (b) QF50, and (c) QF10. The left column represents the image from UCID database and the right column represents the image from UCUS database.

TABLE XVI  
TAMPER DETECTION RESULT FOR 2% SEAM MODIFICATION, AT DIFFERENT QF (UCID IMAGES).

Test	Train	QF10	QF20	QF50	QF75	QF100	Mean
Seam Carving							
QF10	<b>97.94</b>	94.32	83.18	49.81	73.06	79.66	
QF20	88.75	<b>93.80</b>	80.27	51.27	64.57	75.73	
QF50	61.14	64.91	<b>79.90</b>	57.92	53.36	63.45	
QF75	52.77	52.99	56.58	<b>74.70</b>	55.27	58.46	
QF100	79.15	77.09	64.20	64.39	<b>92.94</b>	75.55	
Mean	75.95	76.62	72.83	59.62	67.84	70.57	
Seam Insertion							
QF10	<b>98.02</b>	94.73	84.34	74.14	76.20	85.49	
QF20	89.84	<b>94.54</b>	82.29	69.06	68.31	80.81	
QF50	62.44	66.85	<b>80.90</b>	59.42	54.26	64.77	
QF75	53.18	54.26	56.09	<b>74.25</b>	55.12	58.58	
QF100	76.91	75.00	70.14	80.42	<b>90.55</b>	78.60	
Mean	76.08	77.08	74.75	71.46	68.89	73.65	

TABLE XVII  
TAMPER DETECTION RESULT FOR 20% SEAM MODIFICATION, AT DIFFERENT QF (UCID IMAGES).

Test	Train	QF10	QF20	QF50	QF75	QF100	Mean
Seam Carving							
QF10	<b>96.67</b>	90.77	52.39	44.28	49.51	66.72	
QF20	79.86	<b>90.10</b>	59.90	46.86	49.70	65.28	
QF50	54.41	60.73	<b>80.90</b>	60.54	50.60	61.44	
QF75	51.05	51.94	67.49	<b>82.03</b>	54.30	61.36	
QF100	69.73	53.74	49.36	68.31	<b>98.80</b>	67.99	
Mean	70.34	69.46	62.01	60.40	60.58	64.56	
Seam Insertion							
QF10	<b>98.54</b>	96.00	91.74	87.18	49.25	84.54	
QF20	94.25	<b>96.67</b>	91.29	87.59	49.33	83.83	
QF50	77.65	80.61	<b>90.43</b>	79.48	49.44	75.52	
QF75	58.48	60.61	60.95	<b>86.66</b>	49.78	63.30	
QF100	87.97	85.35	79.78	83.33	<b>98.39</b>	86.96	
Mean	83.38	83.85	82.84	84.85	59.24	78.83	

TABLE XVIII

TAMPER DETECTION RESULT FOR 50% SEAM MODIFICATION, AT DIFFERENT QF (UCID IMAGES).

Test	Train	QF10	QF20	QF50	QF75	QF100	Mean
<u>Seam Carving</u>							
QF10		<b>95.33</b>	85.09	60.99	52.32	49.74	68.69
QF20		84.16	<b>92.75</b>	76.46	61.14	49.96	72.89
QF50		69.13	84.42	<b>93.27</b>	83.48	51.35	76.33
QF75		57.55	71.34	89.87	<b>93.54</b>	55.68	73.60
QF100		47.80	45.78	81.88	91.03	<b>99.63</b>	73.22
<b>Mean</b>		70.79	75.88	80.49	76.30	61.27	72.95
<u>Seam Insertion</u>							
QF10		<b>99.14</b>	97.50	96.08	93.65	49.48	87.17
QF20		97.80	<b>98.84</b>	96.64	95.14	49.48	87.58
QF50		95.40	95.52	<b>97.65</b>	94.36	49.48	86.48
QF75		80.04	81.28	81.84	<b>95.67</b>	49.48	77.66
QF100		93.09	92.94	90.92	89.28	<b>99.40</b>	93.13
<b>Mean</b>		93.09	93.22	92.63	93.62	59.46	86.40

The results of cross validation among QF factors are shown in Tables XVI-XVIII. The result implies that QF in JPEG compression is an impact factor for the symmetric phenomenon in JPEG image. And, this is our concluding finding. However, the relationship among quality factors, blocking effect, and seam tampering rates will be studied in our future work.

In addition, we conducted a small experiment by implementing seam tampering as introduced in [31]. By using our proposed detection method, we found that our detection accuracy is higher than one proposed in [31].

### B. Seam Tamper Localization

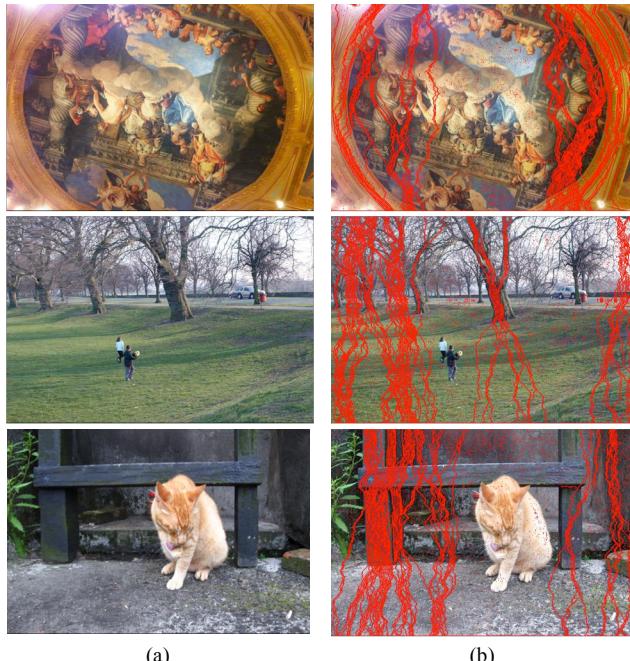


Fig. 14. These figures demonstrate the examples of seam localization in the retargeted images. (a) Tampered images by 10% seam insertion. (b) The localization without referring to the original images.

The examples of retargeting images are demonstrated in Fig. 12. Finding the location of seams is difficult in seam carving since the data is lost from the original image. According to Sarkar *et al.*'s seam insertions detection based on the linear relationship between newly introduce pixels [13], they have described how seam insertion removes a pixel along the seam and replaces it by two pixels, which are averages of pixels lying on/near the seam, as shown in (9). Therefore, for

seam insertion, we can apply the proposed method in [13]. A simple method for detection from every two adjacent pixels after insertion in each iteration is that, we can scan every two adjacent pixels in each row  $j$  to detect the same pixels of  $I(i, j)$  and  $I(i, j+1)$ . The examples of seam insertion localization by the simple detection are demonstrated in Fig. 14. However, this solution can be satisfied in the images, which were modified by seam insertion and resulted in uncompressed images.

For the localization of seam carving, we only use the resulting image, where seams from the original image were already deleted; it is hard to find out where the seams were deleted unless using the original image for a comparison to the seam carving results. The detection of seam insertion resulted in JPEG and the detection of seam carving locations are difficult issues that we are studying and can be considered as our future work. In this research, our contribution is only on high accuracy of tamper detection, instead of localization. In Fig. 14, we used the starting images in QF75 compressed image dataset, and inserted 10% of seams and subsequently resulted in uncompressed images. After that, we located the inserted seams.

### VI. CONCLUSIONS AND FUTURE WORK

Content aware retargeting method by seam modifications can destroy the relation between pixels not only by seam deletion but also by seam insertion. In this research, we proposed a detection method for the tampered JPEG image modified by seam carving or insertion. The relation between pixels in JPEG affects the symmetry of blocking effect that exhibits a symmetrical shape by BACM analysis. The affected significance can be seen, especially in the boundary of BACM. We propose the detection method of tampered JPEG image due to seam modifications. We used a machine learning-based SVM framework to train the models before detecting the modified images. Twenty-two feature vectors were inspired from BACM to find the symmetry of blocking effect. We carefully conducted several experiments. The results illustrate that the accuracy obtained using our proposed method is higher than other existing methods. For interested readers, we have opened our source code and test datasets at <http://video.minelab.tw/DETS/>. For future work, according to [19], lateral aberration could be approximated as an expansion or contraction of the color channels with respect to one another. Seam carving operation also breaks the chromatic aberration in images. Therefore, to get better results in seam modifications detection, we have to consider chromatic aberration. Moreover, our future work will include localization of nearby areas based on the deleted seams with or without using the original images.

In addition, the experimental results demonstrate that the perfect symmetric phenomenon in JPEG compression is found in the images with QF100. The detection results for both seam removal and seam insertion are reasonable and significantly resulted with high accuracy. Whereas, in other QFs, most high accuracies for both seam removal and seam insertion detection

are found in the testing by using the same QF dataset. Although this is the limitation of the proposed method, with different QFs for training and detection, we still demonstrate the feasibility of detection (see our website). We will sincerely take into consideration the analyzing of quantization table and the feasibility of new features in JPEG as our future works.

## REFERENCES

- [1] A.C. Gallagher, "Detection of linear and cubic interpolation in JPEG compressed images," in *Proc. Computer and Robot Vision*, 2005, pp. 65–72.
- [2] A.C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of resampling," *IEEE Trans. Signal Processing*, vol. 53, no. 2, pp. 758–767, Feb., 2005.
- [3] A.C. Popescu and H. Farid, "Statistical tools for digital forensics," in *Lecture Notes in Computer Science*, vol. 3200, 2005, pp. 128–147.
- [4] W.Q. Luo, Z.H. Qu, J.W. Huang and G.P. Qiu, "A novel method for detecting cropped and recompressed image block," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, 2007, pp. II-217-II-220.
- [5] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sept., 1995.
- [6] G. Schaefer and M. Stich, "UCID - An uncompressed colour image database," in *Proc. SPIE, Storage and Retrieval Methods and Applications for Multimedia*, 2004, pp. 472–480.
- [7] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," in *Proc. ACM Transaction on Graphics*, vol. 26, no. 3, 2007, pp. 10(1)–10(8).
- [8] G. Gordon and R. Tibshirani, "Karush-Kuhn-Tucker conditions," in *Optimization*, School of Computer Science, Carnegie Mellon University, 2012.
- [9] M. Kirchner, "Fast and reliable resampling detection by spectral analysis of fixed linear predictor residue," in *ACM Multimedia and Security Workshop*, 2008, pp. 11–20.
- [10] H. Farid, "Image Forgery Detection," in *IEEE Signal Processing Magazine*, 2009, vol. 26, pp. 16–25.
- [11] J. Fridrich, D. Soukal, and J. Lukás, "Detection of copy move forgery in digital images," in *Proc. Digital Forensic Research Workshop*, Cleveland, OH, Aug., 2003.
- [12] A.C. Popescu and H. Farid, "Exposing digital forgeries by detecting duplicated image regions," Dept. Comput. Sci., Dartmouth College, TR2004-515, Sept. 2004.
- [13] A. Sarkar, L. Nataraj, and B.S. Manjunath, "Detection of seam carving and localization of seam insertions in digital images," in *Proc. ACM workshop on Multimedia and Security*, 2009, pp. 107–116.
- [14] Y.Q. Shi, C. Chen, and W. Chen, "A Markov process based approach to effective attacking JPEG steganography," in *Proc. Information Hiding*, 2006, pp. 249–264.
- [15] Z. Fan and R.L. de Queiroz, "Identification of bitmap compression history: JPEG detection and quantizer estimation," in *IEEE Trans. on Image Processing*, vol. 12, no. 2, 2003, pp. 230–235.
- [16] M. Rubinstein, A. Shamir, and S. Avidan, "Improved seam carving for video retargeting," in *ACM Trans. on Graphics*, vol. 27, no. 3, 2008, pp. 16:1–16:9.
- [17] G.K. Wallace, "The JPEG still picture compression standard," in *IEEE Trans. Consumer Electronics*, vol. 38, no. 1, 1992, pp. xviii–xxxiv.
- [18] T. Bianchi and A. Piva, "Image forgery localization via block-grained analysis of JPEG artifacts," in *IEEE Trans. Information Forensics and Security*, vol. 7, no. 3, 2012, pp. 1003–1017.
- [19] M.K. Johnson and H. Farid, "Exposing digital forgeries through chromatic aberration," in *Proc. ACM Multimedia and Security Workshop*, 2006, pp. 48–55.
- [20] W. Lu and M. Wu, "Seam carving estimation using forensic hash," in *Proc. of the thirteenth ACM multimedia workshop on Multimedia and Security*, 2011, pp. 9–14.
- [21] J.D. Wei, Y.J. Lin, Y.J. Wu, "A patch analysis method to detect seam carved images," in *Pattern Recognition Letters* 36, 2014, pp. 100–106.
- [22] X.F. Tao, "Seam Carving," Retrieved November 3, 2014 from <http://cs.brown.edu/courses/cs129/results/proj3/taox/>
- [23] V. Mall, S. Shukla, S.K. Mitra and A.K. Roy, "Comprehensive image index and detection of tampering in a digital image," in *Int. Conf. Informatics, Electronics & Vision (ICIEV)*, 2013, pp. 1–7.
- [24] D. Tralic, J. Petrovic and S. Grgic, "JPEG image tampering detection using blocking artifacts," in *Int. Conf. Systems, Signals and Image Processing (IWSSIP)* 19th, 2012, pp. 5–8.
- [25] "Next generation video: Introducing Daala," Xiph.org, a REDHAT Emerging Technology Project. The page was last modified on 20 June 2013. Retrieved November 2, 2014, from Xiph.org: <https://xiph.org/~xiphmont/demo/daala/demo1.shtml>
- [26] "JPEG," Wikipedia, the free encyclopedia. The page was last modified on 3 November 2014. Retrieved November 2, 2014, from Wikipedia, the free encyclopedia: <http://en.wikipedia.org/wiki/JPEG>
- [27] N. Ponomarenko, K. Egiazarian, V. Lukin, and J. Astola, "Additional lossless compression of JPEG images," in *Proc. of the 4th Intl. Symposium on Image and Signal Processing and Analysis (ISPA 2005)*, Zagreb, Croatia, September 15–17, 2005, pp. 117–120.
- [28] M. Stirner and G. Seelmann, "Improved redundancy reduction for JPEG files," in *Proc. of Picture Coding Symposium (PCS 2007)*, Lisbon, Portugal, November 7–9, 2007.
- [29] W. Dai and T.D. Tran, "Regularity-constrained pre- and post-filtering for block DCT-based systems," *IEEE Transactions on Signal Processing* 51(10), 2003, pp. 2568–2581.
- [30] J. Han, A. Saxena, and V. Melkote, "Jointly optimized spatial prediction and block transform for video and image coding," *IEEE Transactions on Image Processing* (pre-print), Sep. 2011.
- [31] Q. Liu and Z. Chen, "Improved approaches with calibrated neighboring joint density to Steganalysis and seam-carved forgery detection in JPEG images," *ACM Transactions on Intelligent Systems and Technology*, vol. 5, no. 4, Article 63, Publication date: December 2014, pp. 63:1–30.
- [32] I.C. Chang, J.W. Yu, and C.C. Chang, "A forgery detection algorithm for exemplar-based inpainting images using multi-region relation," *Image and Vision Computing*, vol. 31, no. 1, 2013, pp. 57–71.
- [33] Q. Liu, P.A. Cooper, L. Chen, H. Cho, Z. Chen, M. Qiao, Y. Su, M. Wei, and A.H. Sung, "Detection of JPEG double compression and identification of smartphone image source and post-capture manipulation," *Applied Intelligence*, vol. 39, no. 4, 2013, pp. 705–726.
- [34] Y. Chen and C. Hsu, "Detecting recompression of JPEG images via periodicity analysis of compression artifacts for tampering detection," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 2, 2011, pp. 396–406.
- [35] X. Pan and S. Lyu, "Region duplication detection using image feature matching," *IEEE Transactions on Information Forensics and Security* vol. 5, no. 4, 2010, pp. 857–867.
- [36] T. Pevny and J. Fridrich, "Merging Markov and DCT features for multi-class JPEG steganalysis," *Proceedings of SPIE* 6505, 650503, 2007; doi:10.1117/12.696774.



**Kanoksak Wattanachote** is currently reading for his Ph. D. at the Department of Computer Science and Information Engineering, National Central University (NCU), Taiwan. He received his M. S. in Information Management from the School of Advanced Technologies, Asian Institute of Technology (AIT), Thailand and B. S. in Statistics from the Faculty of Science, Srinakharinwirot University (SWU), Thailand. His research interests are in the area of computer vision, image processing, image forensics, JPEG analysis, video editing and interactive multimedia.



**Timothy K. Shih** (<http://www.csie.ncu.edu.tw/~tshih/>) is a Professor at the National Central University, Taiwan. Prof. Shih joined the Educational Activities Board of the IEEE Computer Society. He was the founder and co-editor-in-chief of the International Journal of Distance Education Technologies. He is the Associate Editor of IEEE Computing Now. And, he was the associate editors of

the IEEE Transactions on Learning Technologies, the ACM Transactions on Internet Technology, and the IEEE Transactions on Multimedia. Prof. Shih was the Conference Co-Chair of the 2004 IEEE International Conference on Multimedia and Expo (ICME'2004). Prof. Shih's current research interests include Multimedia Computing and Distance Learning.



**Wen-Lung Chang** received his M. S. in Computer Science and Information Engineering from the Department of Computer Science and Information Engineering, National Central University (NCU), Taiwan in 2013. His research interests are in the area of image processing, image forensics and JPEG analysis.



**Hon-Hang Chang** is currently reading for his Ph. D. at the Department of Computer Science and Information Engineering, National Central University (NCU), Taiwan. He acquired his Master's degree from the Department of Photonics and Communication Engineering, Asia University, Taiwan in 2011. His research interests are in image processing, information hiding and water marking.