Forensically Determining the Order of Signal Processing Operations

Matthew C. Stamm #1, Xiaoyu Chu *2, K. J. Ray Liu *3

Dept. of Electrical and Computer Engineering, Drexel University, Philadelphia, PA, USA
1 mstamm@coe.drexel.edu

* Dept. of Electrical and Computer Engineering, University of Maryland, College Park, MD, USA
2 cxygrace@umd.com, 3 kjrliu@umd.edu

***Abstract*—Currently, many forensic techniques have been developed to determine which processing operations were used to manipulate a multimedia signal. Determining the order in which these operations were applied, however, remains an open challenge. Understanding this order is important because it not only provides greater insight into a signal's processing history, but it can also be used to determine a forger's behavior patterns or provide insight into who manipulated a signal. In this paper, we propose a new forensic detection framework that can be used determine the order in which manipulations were applied to a signal. Additionally, we introduce the notion of a conditional fingerprint to describe how a manipulation's fingerprints can change under subsequent processing. We identify the conditional fingerprints of contrast enhancement followed by resizing, and use our framework to develop an algorithm to determine the order in which resizing and contrast enhancement were applied to an image.**

I. Introduction

In today's society, the majority of multimedia content is captured and distributed in a digital format. The trend towards digitization has greatly increased the ease with which multimedia content can be shared with users all over the world. These gains, however, have come with a price; digital multimedia content can be easily manipulated and falsified. As a result, it is often difficult to trust digital multimedia content. This is problematic because many governmental, legal, and news media organizations rely on multimedia content to make a number of critical decisions.

To combat this situation, researchers have developed a set of digital forensic techniques over the last decade to authenticate digital media content [1]. An important subset of these forensic techniques are designed to identify the use of specific image processing operations and manipulations. These forensic techniques work by detecting the presence of unique artifacts, known as *fingerprints*, left by different signal processing operations. Researchers have designed forensic techniques capable of detecting operations such as contrast enhancement [2], resizing [3], [4], median filtering [5], [6], and multiple JPEG compression [7], [8].

When creating a forgery, it is likely that a forger will make use of multiple editing operations to manipulate a multimedia file. While existing forensic techniques may be able to detect which editing operations were used to create a forgery, *determining the order in which these editing operations were applied* remains an important open problem. Knowledge of the order in which editing operations were applied to a file can provide important information about its processing history. It may also provide insight into a forger's behavior and potentially aid in the identification of the party who created the forgery.

Furthermore, while many forensic techniques are adept at detecting a single specific manipulation, they often encounter difficulties if a forger has applied any subsequent manipulations to a media file. Artifacts left by editing operations that have been applied later in a signal's processing history can potentially alter or disguise fingerprints left by operations that were applied earlier. By understanding the interaction between multiple editing operations, it may be possible to improve the detectability of certain editing operations that have been applied earlier in a signal's processing history.

In this paper, we present a framework for forensically determining the order in which a set of processing operations have been applied to a multimedia signal. We formulate detecting both the presence and order of multiple processing operations that may have been applied to a signal as a multiple hypothesis testing problem. To differentiate between these hypotheses, we introduce the concept of a *conditional fingerprint* to describe how a manipulation's fingerprint changes in the presence of subsequent processing.

After we have presented our framework, we demonstrate its effectiveness by using it to examine the case of images which may have been manipulated using both contrast enhancement and resizing. We identify the conditional fingerprints of these operations and demonstrate how they can be used to improve detection performance and determine the order in which these operations were applied.

II. Order of Operations Detection Framework

Consider the problem of forensically detecting the use of a specific editing operation m_1. When a forensic investigator examines a multimedia signal ψ whose processing history is unknown, they know that ψ must exist in one of two states; ψ has not been processed using m_1 or ψ is a manipulated version of another file ψ'. Typically, an investigator will frame the

WIFS'2013, November 18-21, 2013, Guangzhou, China.
ISBN 978-1-4673-5593-3

detection of the use of m_1 as the following binary hypothesis testing problem

$$\begin{aligned} H_0 &: \psi \text{ is unaltered by } m_1, \\ H_1 &: \psi = m_1(\psi'), \end{aligned} \tag{1}$$

and design a forensic detector δ_1 in the form of a decision rule to distinguish between these hypotheses. This decision rule operates by measuring the strength of some fingerprint ϕ_1 left in ψ by m_1, then comparing this measure to a decision threshold.

If ψ may have additionally been modified by another manipulation m_2, the most widely used detection approach is to frame this problem as another binary hypothesis testing problem in the same form as (1). A detector δ_2 can then be designed to detect the use of m_2 by measuring the strength of the fingerprints ϕ_2 left in ψ by m_2.

This detection approach has several potential drawbacks. One important limitation is that the use of only two binary decision rules is capable of resolving at most only the following four forensic states. By contrast, if an investigator wishes to determine the order in which m_1 and m_2 may have been applied, ψ will actually take on one of *five* forensic states; ψ has not been processed using either m_1 or m_2, ψ has been manipulated using only m_1, ψ has been manipulated using only m_2, ψ has been manipulated first by m_1 then by m_2, or ψ has been manipulated first by m_2 then by m_1.

The disparity between the number possible forensic of states and the number of states that this form of binary hypothesis tests can resolve only grows as the number of possible manipulations considered increases. As a result, this traditional approach to detection, i.e. testing individually for each possible operation, is incapable of resolving the order in which multiple manipulations have occurred.

An additional problem with this approach to detection is that it does not properly account for the possibility that editing operations that occur later in ψ's processing history may cause changes to the fingerprints left by manipulations that occurred earlier in ψ's processing history. These changes may alter important properties of the earlier manipulations' fingerprints, thus making it more difficult or impossible for detectors that measure the strength of unaltered fingerprints to identify these manipulations. For example, δ_1 may yield a positive detection for m_1 if $\psi = m_1(\psi')$ but may yield a negative result if $\psi = m_2(m_1(\psi'))$.

In order to determine the order in which multiple manipulations were applied to multimedia signal and to increase the likelihood of detecting all manipulations applied that signal, we introduce the concept of a manipulation's *conditional fingerprint*. The conditional fingerprint of a manipulation m is defined as the fingerprint caused by m that is present in a signal after that signal has been modified by another operation or ordered sequence of operations. We adopt the notational convention that $\phi_{1|1,2}$ refers to the fingerprint left in a multimedia signal by m_1 when ψ has first been manipulated by m_1 then by m_2, i.e. $\psi = m_2(m_1(\psi'))$. By identifying a manipulation's conditional fingerprints under each possible sequence of manipulations, a forensic investigator can design a set of detectors to differentiate between forensic states that are indistinguishable if only traditional manipulation fingerprints are considered.

To overcome the limitations of performing a set of independent tests for the fingerprints of each individual manipulation, we propose framing manipulation detection as a multiple hypothesis test. When two editing operations, m_1 and m_2, may have been used to manipulate ψ, this test takes the following form:

$$\begin{aligned} H_0 &: \psi \text{ has not been altered by } m_1 \text{ or } m_2, \\ H_1 &: \psi = m_1(\psi'), \\ H_2 &: \psi = m_2(\psi'), \\ H_3 &: \psi = m_2(m_1(\psi')), \\ H_4 &: \psi = m_1(m_2(\psi')). \end{aligned} \tag{2}$$

To differentiate between these states, we propose a "divide-and conquer approach" in which a final forensic decision is made using a sequence of intermediate grouped hypothesis tests. Each intermediate stage searches for the presence of a specific fingerprint or conditional fingerprint in order to reduce the set of candidate states. Depending on the outcome of each intermediate stage, a different test is chosen to be performed at the next stage of detection.

We adopt the notational convention that $H_i^{(j|k_1,\ldots,k_l)}$ represents the i^{th} hypothesis at the j^{th} intermediate stage of detection given that the outcomes of intermediate stages $1,\ldots,l$ were hypotheses $k_1,\ldots,k_l$. For example, $H_0^{(2|1)}$ corresponds to hypothesis H_0 for the 2^{nd} stage of detection given that the outcome of the first stage of detection was $H_1^{(1)}$. Additionally, we define $\delta^{(j|k_1,\ldots,k_l)}$ as the forensic detection rule applied at the j^{th} intermediate stage of detection given that the outcomes of intermediate stages $1,\ldots,l$ were hypotheses $k_1,\ldots,k_l$.

To provide a brief example of what one stage of this form of detection would look like, suppose m_1 has little effect on the fingerprints of m_2 so that $\phi_{2|2,1} = \phi_2$. Additionally, let us also suppose that m_2 changes the fingerprints left by m_1 so that $\phi_{1|1,2} \neq \phi_1$. In this scenario, a forensic investigator would design the first stage of detection to determine if ψ was modified using m_2 by distinguishing between $H_0^{(1)} = \{H_0, H_1\}$ and $H_1^{(1)} = \{H_2, H_3, H_4\}$. Subsequent stages would use different detectors to determine if ψ was modified using m_1 depending on the outcome of the first stage.

In the remainder of this paper, we apply this framework to the problem of detecting contrast enhancement and resizing in digital images. We use these two operations to provide concrete examples of conditional fingerprints and of how our framework is used to design a detection algorithm to determine the order in which two operations were applied. Furthermore, we show that the conditional fingerprints of contrast enhancement can be used to detect contrast enhancement under conditions that previously appeared problematic.

III. Existing Approaches to Contrast Enhancement and Resizing Detection

In this section, we provide a brief review of existing approaches to both contrast enhancement and resizing detection.

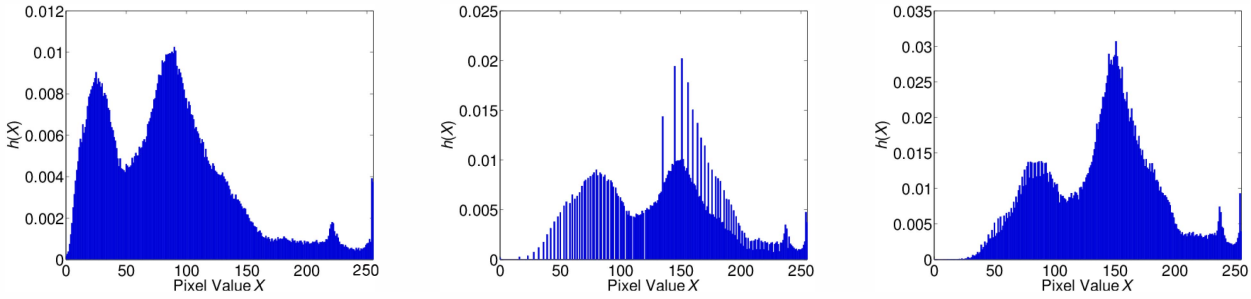

Fig. 1. Pixel value histograms of (a) an unaltered image, (b) a contrast enhanced image, and (c) an image that has been contrast enhanced then resized.

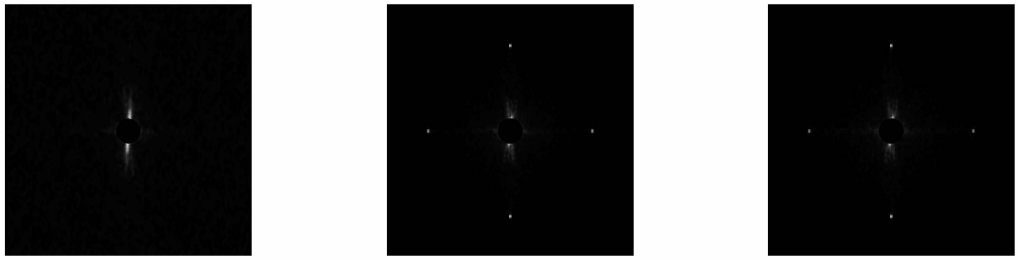

Fig. 2. The magnitude of the DFT of the p-map from (a) an unaltered image, (b) an image that has been resized by a factor of 1.5, and (c) an image that has been resized by a factor of 1.5 then contrast enhanced. Note the bright spectral peaks corresponding to resizing fingerprints in (b) and (c).

A. Contrast Enhancement

Contrast enhancement operations work by applying a non-decreasing nonlinear mapping to the pixel values of an image. Locally contractive regions of these mappings will cause certain sets of distinct input pixel values to map to the same output value. This will introduce impulsive peaks in the histogram of a contrast enhanced image at the pixel value locations where two input values are mapped to the same output value. Similarly, locally expansive regions of these mappings will cause two adjacent pixel values to map to output values separated by at least one intermediate value. This will result in sudden zeros or gaps in a contrast enhanced image's pixel value histogram. These impulsive peaks and gaps in a contrast enhanced image's pixel value histogram are the standard fingerprints left by contrast enhancement [2]. An example of contrast enhancement fingerprints can be seen in Fig. 1.

The strength of these fingerprints can be easily measured using a frequency domain representation of an image's normalized pixel value histogram [2]. Since an unaltered image's pixel value histogram is typically smooth, the majority of the energy in its Fourier transform will be concentrated in low frequency regions. By contrast, contrast enhancement fingerprints will result in significant high frequency content due to their impulsive nature. Contrast enhancement detection is performed using a frequency domain measure of these fingerprints according to the following procedure.

First, an image's normalized pixel value histogram $h(x)$ is multiplied by a 'pinchoff function' $p(x)$ to obtain the modified histogram [2]

$$g(x) = h(x)p(x). \tag{3}$$

This is done to suppress any histogram artifacts caused by high-end or low-end saturation that may interfere with detection. Next, a measure of the energy in the high frequency components of the modified histogram is computed using the equation

$$F = \sum_{k} |\beta(k)G(k)|, \tag{4}$$

where $G(k)$ is the DFT of $g(x)$ and $\beta(k)$ is a frequency weighting function. A commonly used weighting function is $\beta(k) = 1$ for $c \leq k \leq 128$ and 0 elsewhere, where c is a user specified frequency cutoff.

Finally, an image is classified as unaltered or contrast enhanced using the decision rule

$$\delta_{ce} = \begin{cases} \text{not contrast enhanced} & \text{if } F < \tau_{ce}, \\ \text{contrast enhanced} & \text{if } F \geq \tau_{ce}, \end{cases} \tag{5}$$

where τ_{ce} is a decision threshold.

B. Resizing

Image resizing is performed by first determining a new image sampling grid, then by interpolating values on this grid that are not directly observed. Typically, interpolation is performed using a linear operator. Several techniques have been developed to perform resizing detection, however the most widely used techniques are based off of the observation that when an image is resized, interpolated pixels will be more highly correlated with their neighbors than directly observed pixels [3].

Initial work in resizing detection operates by using the Expectation-Maximization (EM) algorithm to jointly estimate a linear predictor for each pixel value along with the probability that a pixel is correlated with its neighbors [3]. If an image

has been resized by a rational factor, this set of probabilities, known as a p-map, will be periodic. While this approach to detection is effective, the EM algorithm is computationally expensive to implement.

To overcome this shortcoming, a computationally efficient alternative approach toward estimating the p-map has been developed [4]. This approach obtains the p-map by first predicting the value of each pixel using a predetermined linear filter α. The prediction residual e is then determined by subtracting each predicted pixel value from the true value and used to calculate the p-map p according to the equation

$$p_{i,j} = \lambda \exp\left(-|e_{i,j}|^{\mu}/\sigma\right), \tag{6}$$

where λ, $\mu \geq 1$, and $\sigma > 0$ are controlling parameters. If an image has been resampled, distinct spectral peaks will be present in the 2D DFT of the p-map, as can be seen in Fig. 2.

Resizing detection is performed by first using the 2D DFT of p to calculate the cumulative periodogram C of the p-map. A detection statistic ρ is calculated using the equation

$$\rho = \max_{k_1,k_2} |\nabla C(k_1, k_2)|, \tag{7}$$

and resizing detection is performed according to the following decision rule

$$\delta_{rs} = \begin{cases} \text{not resized} & \text{if } \rho < \tau_{rs}, \\ \text{resized} & \text{if } \rho \geq \tau_{rs}, \end{cases} \tag{8}$$

where τ_{rs} is a decision threshold.

IV. Conditional Fingerprints of Resizing and Contrast Enhancement

While techniques have been designed to detect both contrast enhancement and resizing, little work has examined the effect that subsequent operations other than JPEG compression will have on the fingerprints of these manipulations.

Fig. 2(c) shows the DFT of the p-map of an image which has undergone resizing followed by a form of contrast enhancement known as gamma correction. The spectral peaks corresponding to resizing fingerprints can still clearly be seen in this figure, thus suggesting contrast enhancement has little effect on resizing fingerprints. As we can see from this figure, contrast enhancement has essentially no effect on existing resizing fingerprints. Therefore, if we let rs represent resizing and ce represent contrast enhancement, we can state that $\phi_{rs|rs,ce} = \phi_{rs}$. This means that resizing fingerprints are independent of subsequent contrast enhancement operations.

By contrast, Fig. 1(c) shows the normalized pixel value histogram of an image that has undergone contrast enhancement followed by resizing. From this figure, we can see that resizing has caused contrast enhancement fingerprints to remain largely absent from the image's pixel value histogram. As a result, images that have been resized after they have undergone contrast enhancement will appear to the detector δ_{ce} given in (5) as if they had not been contrast enhanced at all. We can clearly see that the conditional fingerprints of contrast enhancement followed by resizing are not the same as the standard fingerprints of contrast enhancement.

To identify the conditional fingerprints of contrast enhancement followed by resizing, we must first closely examine how resizing alters an image. When a digital image is captured, each pixel value corresponds to a sample of the illumination intensity at that pixel's spatial location. If the image is subsequently resized, a new sampling grid is formed and values are assigned to the pixels at each of these new locations. These locations are determined by the scaling factor s, which for the purposes of this paper we assume can be expressed as a rational number $s = n/d$ such that $n, d \in \mathbb{N}$. If we adopt the convention that each pixel in the original image is spaced one unit apart, then the pixels in the resized image will be spaced d/n units apart.

Once the new pixel locations are determined, interpolation is used to calculate the pixel values at these locations. It can be shown, however, that every d^{th} pixel in the resized image will occur at the same spatial location as a pixel in the original image. As a result, the value of each of these pixels will directly correspond to the value the pixel at the same spatial location in the image before it was resized. If an image $\psi = rs(\psi')$, we can form the set ξ of pixels in a resized image ψ that directly correspond to pixels in ψ' as

$$\xi = \{\psi_{i,j} | i \bmod n = 1, j \bmod n = 1\}, \tag{9}$$

where $\bmod$ is the modulo operator.

If we now suppose that $\psi' = ce(\psi'')$ so that $\psi = rs(ce(\psi''))$, we can see that each pixel in ξ will correspond to a pixel in the contrast enhanced version of ψ''. As a result, traditional contrast enhancement fingerprints will be present in the normalized pixel value histogram of ξ.

The reason that we do not observe contrast enhancement fingerprints in the pixel value histogram of ψ as a whole is that all pixels in ψ that are not part of ξ are interpolated. Since interpolation is performed by computing a linear combination of all of the pixels in ψ' that fall within the interpolation filter window, each interpolated pixel value could potentially lie anywhere in the convex hull of these pixel values. This effectively smooths the histogram of ψ, thus masking contrast enhancement fingerprints. Additionally, this will cause the normalized pixel value histogram of ξ to differ significantly from the normalized pixel value histogram of ψ since ξ will contain contrast enhancement fingerprints that are not present in ψ as a whole.

Using this information, we can characterize the conditional fingerprints $\psi_{ce|ce,rs}$ of contrast enhancement followed by resizing as the presence of impulsive peaks and gaps in the normalized pixel value histogram of the set ξ coupled with a significant difference between the normalized pixel value histograms of ξ and ψ as a whole. We can now use these fingerprints to perform improved contrast enhancement detection and to determine the order in which contrast enhancement and resizing has been applied to an image.

V. Determining the Order of Contrast Enhancement and Resizing

Now that we know the conditional fingerprints $\phi_{rs|rs,ce}$ and $\phi_{ce|ce,rs}$, we can use the framework that we proposed

in Section II to design an algorithm to detect both contrast enhancement and resizing along with the order in which they were applied to an image ψ. The first step in this process is to formulate our problem as the following multiple hypothesis testing problem

$$\begin{aligned} H_0 &: \psi \text{ has not been altered using contrast} \\ &\quad \text{enhancement or resizing,} \\ H_1 &: \psi = rs(\psi'), \\ H_2 &: \psi = ce(\psi'), \\ H_3 &: \psi = rs(ce(\psi')). \\ H_4 &: \psi = ce(rs(\psi')), \end{aligned} \tag{10}$$

Since contrast enhancement fingerprints are masked by the subsequent use of resizing but not vice versa, we use the first stage of detection to determine if ψ has undergone resizing at any point. To do this, we choose our grouped hypotheses at stage 1 of our detection algorithm to be $H_0^{(1)} = \{H_0, H_2\}$ and $H_1^{(1)} = \{H_1, H_3, H_4\}$. Because resizing fingerprints are independent of subsequent applications of contrast enhancement, we can use δ_{rs} to differentiate between these hypotheses. As a result, we use $\delta^{(1)} = \delta_{rs}$.

If resizing fingerprints are detected in the first stage (i.e. $\delta^{(1)}$ yields the result $H_1^{(1)}$), we choose our next set of hypotheses as $H_0^{(2|1)} = \{H_1, H_4\}$ and $H_1^{(2|1)} = \{H_3\}$. This is equivalent to testing specifically for the conditional fingerprints $\phi_{ce|ce,rs}$ left by contrast enhancement followed by resizing. We have intentionally chosen to separate out hypothesis H_3 at this stage because in certain instances, images that have undergone contrast enhancement followed by resizing (H_3) will still have pixel value histograms containing enough high frequency content to be misclassified as images that have undergone resizing followed by contrast enhancement (H_4).

To differentiate between hypotheses $H_0^{(2|1)}$ and $H_1^{(2|1)}$, we first calculate the normalized pixel value histogram h of ψ along with the normalized pixel value histogram h_ξ of the set ξ. We obtain a measure F_ξ of the strength of contrast enhancement fingerprints in ξ by substituting h_ξ for h in (3) and (4). Additionally, we measure the distance ν between the normalized histograms h and h_ξ according to the equation

$$\nu = \sum_x |h(x) - h_\xi(x)|^2. \tag{11}$$

We then use the following decision rule to differentiate between each hypothesis at this stage

$$\delta^{(2|1)} = \begin{cases} H_0^{(2|1)} & \text{otherwise,} \\ H_1^{(2|1)} & \text{if } F_\xi \geq \tau_{(2|1)} \text{ and } \nu \geq \eta, \end{cases} \tag{12}$$

where $\tau_{(2|1)}$ and η are decision thresholds. We note that by making use of the conditional fingerprint $\phi_{ce|ce,rs}$, this stage of our algorithm is able to detect the use of contrast enhancement under conditions that previously would have likely resulted in a missed detection.

If the result of $\delta^{(2|1)}$ yields $H_1^{(2|1)}$, our detection algorithm terminates and returns a result of H_3. Otherwise, our algorithm proceeds on to stage 3. Given that the results of the previous two stages were $H_1^{(1)}$ and $H_0^{(2|1)}$, the only remaining fingerprints that we must test for are traditional contrast enhancement fingerprints. By choosing our hypotheses at stage 3 as $H_0^{(3|1,0)} = H_1$ and $H_0^{(3|1,0)} = H_4$, δ_{ce} can be used to differentiate between these hypotheses. As a result, we assign $\delta^{(3|1,0)} = \delta_{ce}$. Regardless of the result yielded by $\delta^{(3|1,0)}$, our algorithm terminates at this stage. Hypothesis H_4 is returned if contrast enhancement is detected at this stage and hypothesis H_1 is returned otherwise.

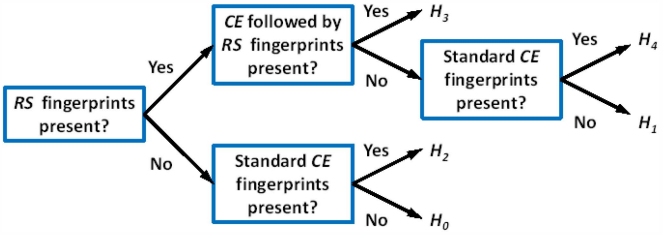

Fig. 3. Flow chart of our detection algorithm.

If resizing fingerprints are not detected in the first stage (i.e. $\delta^{(1)}$ yields the result $H_0^{(1)}$), then the only way that ψ may have been manipulated is using contrast enhancement alone. Since the only fingerprints that may be present are standard contrast enhancement fingerprints, we define our hypotheses at this stage as $H_0^{(2|0)} = H_0$ and $H_1^{(2|0)} = H_2$ and use the decision rule $\delta^{(2|0)} = \delta_{ce}$ to differentiate between them. Hypothesis H_2 is returned if contrast enhancement is detected at this stage and hypothesis H_0 is returned otherwise.

A flow chart outlining our entire detection algorithm is shown in Fig. 3.

VI. Experimental Results

In order to demonstrate our detection framework's ability to determine the order in which contrast enhancement and resizing operations were applied to an image, we created a database of unaltered grayscale images from the 1338 images in the Uncompressed Color Image Database [9]. We then used these images to conduct a set of experiments designed to evaluate our detection algorithm's performance.

In our first experiment, we created a set of contrast enhanced images by applying gamma correction with $\gamma = 0.5$ to each unaltered image, along with a set of resized images created by using bilinear interpolation to scale the unaltered images by a factor of 1.5. Next we created two additional sets of images by applying gamma correction to the resized images and resizing to the gamma corrected images. This resulted in a total of 6690 images whose processing history matched one of the five forensic states described in (10).

After creating this set of testing images, we classified each image as 'unaltered', 'resized', 'contrast enhanced', 'resized then contrast enhanced', or 'contrast enhanced then resized' using the sequence of grouped hypothesis testing algorithm that we described in Section VI. We repeated these test while varying the decision thresholds in each step over a broad range of values, then aggregated the detection results and used them

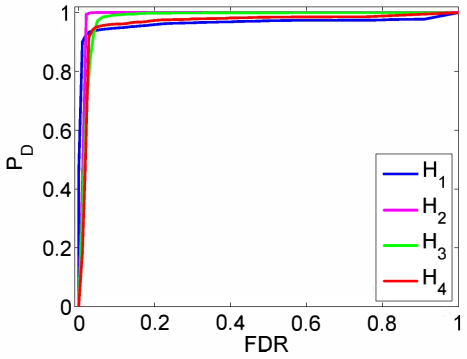

Fig. 4. Detection results when testing for a combination of gamma correction with $\gamma = 0.5$ resizing via bilinear interpolation using a scaling factor of 1.5. Definitions of each hypothesis can be found in (10).

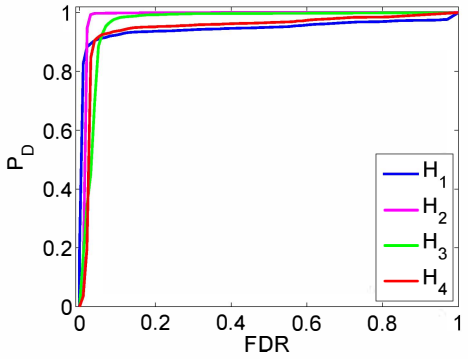

Fig. 5. Detection results when testing for a combination of gamma correction with $\gamma = 0.7$ resizing via bilinear interpolation using a scaling factor of 1.25. Definitions of each hypothesis can be found in (10).

to calculate the probability of detection P_D for each hypothesis H_i. This was done by summing the number of images that were correctly classified as belonging to hypothesis H_i and dividing this quantity by the total number of images for which hypothesis H_i was true. Additionally, we calculated the corresponding false discovery rate FDR for each hypothesis H_i by summing the number of images for which our detector yielded an incorrect decision of H_i and dividing by the total number of images that resulted in detections for H_i.

Using these results, we constructed the set of curves displayed in Fig. 4 which plot the P_D against the FDR for each alternative hypothesis. From the calculation of the FDR, we can see that it is somewhat analogous to the probability of false alarm. As a result, these curves resemble ROC curves for each hypothesis. From Fig. 4, we can see that we were able to achieve a $P_D > 94\%$ at a $FDR = 5\%$ for each hypothesis, including hypotheses H_3 and H_4 which correspond to using both contrast enhancement and resizing, but in different orders.

To further test our framework, we repeated this experiment using $\gamma = 0.7$ when performing gamma correction and a scaling factor of 1.25 when resizing each image. This choice of γ and the scaling factor will result in weaker contrast enhancement fingerprints and rescaling fingerprints than those present in our previous experiment. After using our framework to test for both operations and determine the order in which they were applied, we aggregated our detection results and used them to create the curves shown in Fig. 5. In this experiment,we were able to achieve a $P_D > 91\%$ at a $FDR = 5\%$ for each hypothesis.

The results of both of these experiments demonstrate that our framework can be used to accurately determine the order in which these manipulations were applied. Furthermore, our framework allowed us to accurately detect the use of contrast enhancement, even when it was followed by resizing. This would not be possible using standard detection approaches because resizing fingerprints mask standard contrast enhancement fingerprints. From these results, however, we see that contrast enhancement can be reliably detected using its conditional fingerprint $\phi_{ce\,|\,ce,rs}$.

VII. Conclusion

In this paper, we have proposed a new forensic detection framework that can be used to determine the order in which manipulations were applied to a signal. This is done by posing detection as a multiple hypothesis test and using a sequence of grouped hypothesis tests to differentiate between each ordered pair of manipulations. Furthermore, we have introduced the notion of a conditional fingerprint to describe how a manipulation's fingerprints can change under subsequent processing. We have identified the conditional fingerprints of contrast enhancement followed by resizing, and used these fingerprints in conjunction with our framework to develop an algorithm to detect the order in which resizing and contrast enhancement were used to manipulate an image. Through a series of experiments, we have demonstrated the effectiveness of our algorithm and shown that it can be used to detect contrast enhancement in conditions that were standard approaches to detection view as very unfavorable.

References

[1] M. C. Stamm, M. Wu, and K. J. R. Liu, "Information forensics: An overview of the first decade," *IEEE Access*, vol. 1, pp. 167–200, 2013.

[2] M. C. Stamm and K. J. R. Liu, "Forensic detection of image manipulation using statistical intrinsic fingerprints," *IEEE Trans. Information Forensics and Security*, vol. 5, no. 3, pp. 492 –506, Sep. 2010.

[3] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of resampling," *IEEE Trans. on Signal Processing*, vol. 53, no. 2, pp. 758 – 767, Feb. 2005.

[4] M. Kirchner, "Fast and reliable resampling detection by spectral analysis of fixed linear predictor residue," in *ACM workshop on Multimedia and Security*, Oxford, United Kingdom, 2008, pp. 11–20.

[5] M. Kirchner and J. Fridrich, "On detection of median filtering in digital images," *SPIE Electronic Imaging: Security, Steganography, and Watermarking of Multimedia Contents*, 2010.

[6] X. Kang, M. C. Stamm, A. Peng, and K. J. R. Liu, "Robust median filtering forensics based on the autoregressive model of median filtered residual," in *APSIPA Annual Summit & Conference*, Dec. 2012, pp. 1 –9.

[7] T. Pevný and J. Fridrich, "Detection of double-compression in JPEG images for applications in steganography," *IEEE Trans. on Information Forensics and Security*, vol. 3, no. 2, pp. 247 –258, Jun. 2008.

[8] T. Bianchi and A. Piva, "Image forgery localization via block-grained analysis of JPEG artifacts," *IEEE Trans. on Information Forensics and Security*, vol. 7, no. 3, pp. 1003 –1017, Jun. 2012.

[9] G. Schaefer and M. Stich, "UCID: an uncompressed color image database," in *Proc. SPIE: Storage and Retrieval Methods and Applications for Multimedia*, vol. 5307, no. 1, 2004, pp. 472–480.