Contrast Enhancement Estimation for Digital Image Forensics

LONGYIN WEN, GE Global Research Center, USA
HONGGANG QI, University of Chinese Academy of Sciences, China
SIWEI LYU, University at Albany, State University of New York, USA and Tianjin Normal University, China

Inconsistency in contrast enhancement can be used to expose image forgeries. In this work, we describe a new method to estimate contrast enhancement operations from a single image. Our method takes advantage of the nature of contrast enhancement as a mapping between pixel values, and the distinct characteristics it introduces to the image pixel histogram. Our method recovers the original pixel histogram and the contrast enhancement simultaneously from a single image with an iterative algorithm. Unlike previous works, our method is robust in the presence of additive noise perturbations that are used to hide the traces of contrast enhancement. Furthermore, we also develop an effective method to detect image regions undergone contrast enhancement transformations that are different from the rest of the image, and use this method to detect composite images. We perform extensive experimental evaluations to demonstrate the efficacy and efficiency of our method.

CCS Concepts: • **Mathematics of computing** → *Convex optimization*; • **Computing methodologies** → *Scene anomaly detection*; • **Applied computing** → *Evidence collection, storage and analysis*;

Additional Key Words and Phrases: Media Forensics, Contrast Enhancement, Pixel Histogram

ACM Reference Format:
Longyin Wen, Honggang Qi, and Siwei Lyu. 2017. Contrast Enhancement Estimation for Digital Image Forensics. *ACM Trans. Multimedia Comput. Commun. Appl.* 0, 0, Article 00 (2017), 18 pages. https://doi.org/0000001.0000001

1 INTRODUCTION

The integrity of digital images has been challenged by the development of sophisticated image editing tools (e.g., Adobe Photoshop), which can modify contents of digital images with minimal visible traces. Accordingly, the research field of digital image forensics [11] has experienced rapid developments in the past decade. Important cues to authenticate digital images and detect tampering can be found from various steps in the image capture and processing pipeline, and one of such operations is contrast enhancement. Contrast enhancement is a nonlinear monotonic function of pixel intensity, and it is frequently exploited to enhance image details of over-or under-exposed regions. Commonly used contrast enhancement transforms include gamma correction, sigmoid

Authors' addresses: Longyin Wen, GE Global Research Center, 1 Research Circle, Niskayuna, NY, 12309, USA; Honggang Qi, University of Chinese Academy of Sciences, School of Computer and Control Engineering, No.3, ZhongGuanCun NanYiTiao, Beijing, 100049, China; Siwei Lyu, University at Albany, State University of New York, Computer Science Department, 1400 Washington Avenue, Albany, NY, 12222, USA, Tianjin Normal University, School of Computer and Information Engineering, No.393, Binshui XiDao, Tianjin, China.

1551-6857/2017/0-ART00 $15.00
https://doi.org/0000001.0000001

stretching and histogram equalization [13]. In digital image forensics, recovering contrast enhancement is useful to reconstruct the processing history of an image. Also, detecting regions undergone different contrast enhancement can be used to expose a composite image.

There have been several methods to estimate contrast enhancement from an image [5, 6, 10, 16, 19, 20, 23]. However, these methods have two main limitations. First, most of these algorithms are designed for a specific type of contrast enhancement transform (*e.g.*, gamma correction). Second, these algorithms in general lack robustness with regards to noise perturbations that are added to hide the traces of contrast enhancement [1, 7].

In this work, we describe a general method to recover contrast enhancement from a single image. Our method exploits the observations that: (i) although contrast enhancement is typically a nonlinear function of pixel values, it is a linear transformation of pixel histogram; (ii) pixel histogram after contrast enhancement tends to have more empty bins; and (iii) the effect of additive noise corresponds to a convolution of pixel histogramwith the noise distribution. Accordingly, we formulate the estimation of contrast enhancement as an optimization problem seeking recovered pixel histogram to be consistent with the observed pixel histogram after contrast enhancement transform is applied, while with minimum number of empty bins. The original problem is intractable, so we further provide a continuous relaxation and an efficient numerical algorithm to solve the relaxed problem. Our formulation can handle the estimation of parametric and nonparametric contrast enhancement transforms, and is robust to additive noise perturbations. Furthermore, we also develop an effective method to detect regions undergone contrast enhancement operation different from the remaining of the image, and use this method to detect composite images generated by splicing.

A preliminary version of this work was published in [22]. The current work extends our previous method in several key aspects. First, this work uses a more stable property of contrast enhancement on pixel histogram based on the number of empty bins, which leads to a new type of regularizer in the optimization objective. Furthermore, we include a new optimization method based on the Wasserstein distance and augmented the original algorithm to handle additive noises. The optimization of the overall problem is solved with a more efficient projected gradient descent method.

The rest of this paper is organized as follows. In Section 2, we review relevant previous works. In Section 3, we elaborate on the relation of contrast enhancement and pixel histogram, and describe our algorithm estimating parametric and nonparametric contrast enhancement transforms. In section 4, we present the experimental evaluations of the global contrast enhancement estimation algorithm. Section 5 focuses on a local contrast enhancement estimation algorithm based on the global contrast enhancement estimation algorithm and graph cut minimization. Section 6 concludes the article with discussion and future works.

2 BACKGROUND AND RELATED WORKS

2.1 Contrast Enhancementas Linear Operator on Pixel Histogram

Our discussion is for gray-scale images of b bit-pixels. A (normalized) pixel histogram represents the fractions of pixels taking an individual value out of all 2^b different grayscale values, and is usually interpreted as the probability distribution of a random variable X over $\{0, \cdots, n = 2^b - 1\}$.

A contrast enhancement is a point-wise monotonic transform between pixel values $i, j \in \{0, \cdots, n\}$, defined as $i = \phi(j) := [m(i)]$, where $m(\cdot) : [0, n] \mapsto [0, n]$ is a continuous non-decreasing function, and $[\cdot]$ is the rounding operation that maps a real number to its nearest integer.

There are two categories of contrast enhancement transforms. A *parametric* contrast enhancement transform can be determined with a set of parameters. An example of parametric contrast enhancement transform is *gamma correction*,

$$j = \phi_\gamma(i) := \left\lceil n\left(\frac{i}{n}\right)^\gamma \right\rfloor, \tag{1}$$

where $\gamma \geq 0$ is the parameter controlling the shape of the transform. Another often-used parametric contrast enhancement transform is *sigmoid stretching*,

$$j = \phi_{\alpha,\mu}(i) := \left\lceil n\left(\frac{S\left(\frac{i-n\mu}{n\alpha}\right) - S\left(\frac{-n\mu}{n\alpha}\right)}{S\left(\frac{n(1-\mu)}{n\alpha}\right) - S\left(\frac{-n\mu}{n\alpha}\right)}\right)\right\rfloor, \tag{2}$$

where $\alpha > 0$ and $\mu \in [0,1]$ are two parameters, and $S(x) = \frac{1}{1+\exp(-x)}$ is the sigmoid function. On the other hand, a *nonparametric* contrast enhancement transform affords no simple parametric form and has to be specified for all $i \in \{0, \cdots, n\}$. An example of the nonparametric contrast enhancement transform is *histogram equalization*, which maps the pixel histogram of an image to match a uniform distribution over $\{0, \cdots, n\}$.

Consider two images I and $\tilde{I}$, with the pixels of $\tilde{I}$ obtained from those of I using a contrast enhancement transform ϕ. We introduce two random variables $X, Y \in \{0, \cdots, n\}$ as the pixels of I and $\tilde{I}$, hence $Y = \phi(X)$. Using the probability interpretation of pixel histogram, the contrast enhancement transform between X and Y induces the conditional probability distribution $\Pr(Y = j|X = i) = \mathbf{1}_{j=\phi(i)}$, where $\mathbf{1}_c$ is the indicator function whose output is 1 if c is true and zero otherwise. As such, we have

$$\Pr(Y = j) = \textstyle\sum_i \mathbf{1}_{j=\phi(i)} \Pr(X = i). \tag{3}$$

We can model a pixel histogram as a vector on the n-dimensional probability simplex, *i.e.*, $\mathbf{h} \in \Delta^{n+1} := \{\mathbf{h}|\mathbf{h} \geq 0, \mathbf{1}^T\mathbf{h} = 1\}$ with $h_{i+1} = \Pr(X = i)$ for $i = 0, \cdots, n$, and a contrast enhancement ϕ as an $(n+1) \times (n+1)$ matrix $T_\phi : \left(T_\phi\right)_{i+1,j+1} = \mathbf{1}_{j=\phi(i)}$. Note that T has nonnegative entries, and each column sum to one, *i.e.*, $T^\top\mathbf{1} = \mathbf{1}$. With vector $\mathbf{h}$ and matrix T_ϕ, we can rewrite Eq. (3) as

$$\tilde{\mathbf{h}} = T_\phi \mathbf{h}, \tag{4}$$

i.e., the pixel histograms of I and $\tilde{I}$ are related by a *linear* transform, even though ϕ is a nonlinear function of X.

2.2 Previous Works

Except for the case of identity, a contrast enhancement will map multiple input values to a single output value (correspondingly, there will be values to which no input pixel value maps), a result from the *pigeonhole principle* [14]. This leaves "peaks and gaps" in the pixel histogram after a contrast enhancement transform is applied, which have inspired several works to develop statistical features to detect the existence of contrast enhancement in an image. The works in [19, 20] describe an iterative algorithm to jointly estimate a gamma correction, based on a probabilistic model of pixel histogram and an exhaustive matching procedure to determine which histogram entries are most likely to correspond to artifacts caused by gamma correction. The statistical procedure of [19] is further refined in [16] to determine if an image has undergone gamma correction or histogram equalization. However, all these methods aim only to detect the existence of certain contrast enhancement in an image, but not to recover the actual form of the contrast enhancement function.

There are several methods that can also recover the functional form of contrast enhancement. The method in [10] recovers gamma correction from an image using bi-spectra. The method of [5] uses the features developed in [20] to recover the actual gamma value by applying different gamma values to a uniform histogram and identifying the optimal value that best matches the observed pixel histogram features. This work is further extended in [6] to recover contrast enhancement of a JPEG compressed image. Our previous work [23] uses the increased non-smoothness of pixel histogram to recover the corresponding contrast enhancement transform. However, most of the previous methods require knowledge of the type of contrast enhancement *a priori*. Another common

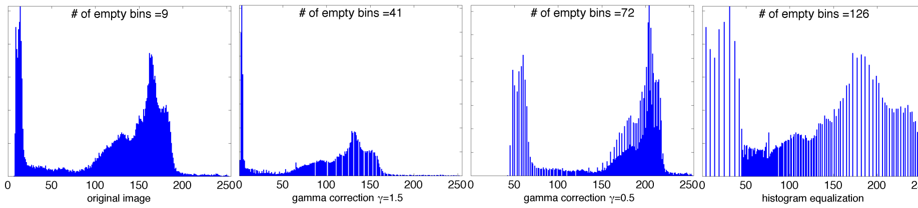

Fig. 1. *Effects of contrast enhancement on the pixel histogram. The four panels correspond to the pixel histograms of (i) the original image, (ii) the image after a gamma correction with* $\gamma = 1.5$*, (iii) the image after a gamma correction with* $\gamma = 0.5$*, and (iv) the image after histogram equalization, each panel is marked with the number of empty bins in the corresponding pixel histogram.*

problem of these previous methods is that they are not robust with regards to additive noise that is added to intentionally hide the trace of contrast enhancement[1, 7].

Methods for local contrast enhancement detection have also been studied in several previous works[6, 16, 19]. However, most of these works are limited in that they can only recover at the level of image blocks of large sizes. The high computational cost is one reason why existing methods cannot be used for *pixel-level* localization of regions with different contrast enhancement transforms.

3 METHOD

In this section, we describe our method for contrast enhancement estimation from an image. We start with a property of the pixel histogram of contrast enhancement transformed image. This is then utilized to formulate the contrast enhancement estimation problem as an optimization problem, and we further provide details of the solution algorithm.

3.1 Pixel Histogram After Contrast Enhancement

The effect of a contrast enhancement transform is a redistribution of pixel values in an image. In particular, the number of empty bins in the pixel histogram does not decrease after a contrast enhancement transform is applied to an image. This is formally described in the following result.

Theorem 1. *Define* $\Omega(\mathbf{h}) = \sum_{i=1}^{n} \mathbf{1}_{h_i=0}$, i.e., $\Omega(\mathbf{h})$ *counts the number of empty bins in* $\mathbf{h}$*, then we have* $\Omega(\mathbf{h}) \leq \Omega(T_\phi \mathbf{h})$.

Proof of this result can be found in the Appendix. Albeit this is a simple observation, it holds regardless of the image or the contrast enhancement transform. Fig.1 demonstrate this effect using an 8-bit grayscale image as an example. The four panels of Fig.1 correspond to the pixel histograms of (i) the original image, (ii) the image after a gamma correction with $\gamma = 1.5$, (iii) the image after a gamma correction with $\gamma = 0.5$, and (iv) the image after histogram equalization, each with the number of empty bins annotated. Note the significant increment of the number of empty bins in the contrast enhancement transformed images (*e.g.*, the number of empty bins increases from 9 to 126 in the case of histogram equalization).

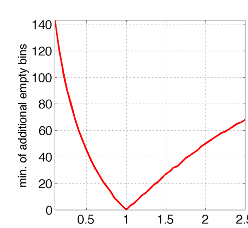

Fig. 2. *The minima of changes in the number of empty bins over* 2000 *natural images after gamma corrections with* γ *value in the range of* $\{0.1 : 0.01 : 2.5\}$ *are applied. Note the prominent trend of increasing number of empty bins after contrast enhancement is applied to these images. See more details in the texts.*

Fig.2 corresponds to a quantitative evaluation. Specifically, we choose 2,000 natural images from RAISE dataset[9][1], and apply gamma corrections with γ value in the range of $\{0.1 : 0.01 : 2.5\}$

[1]The original images are in the 12-bit or 14-bit uncompressed or lossless compressed NEF or TIFF format. We downloaded the full RAISE dataset but use a random subset of 2,000 images for testing our algorithm. We use the green channel of the RGB color image as in [6]. The pixel histograms are vectors of $2^{12} = 4,096$ and $2^{14} = 16,384$ dimensions, respectively.

to each image. We then compute the *difference* between the number of empty bins of the gamma corrected image with that of the original image (therefore it is always zero for $\gamma = 1$ which corresponds to the original image). We then show the *minima* of these differences over the 2,000 images in Fig.2. Note that these minima are positive, indicating a prominent trend of increasing number of empty bins after contrast enhancement transform is applied.

3.2 Effect of Additive Noise

The change in the number of empty bins of pixel histogram caused by contrast enhancement may be obscured by adding noise to the contrast enhancement transformed image (see Fig.3), a fact employed in recent anti-forensic techniques aiming at hiding the trace of contrast enhancement[1, 7]. However, this artifact introduced by additive noise can be precisely modeled using the same mathematical framework. Consider two random variables $X, Y \in \{0, \cdots, n\}$ corresponding to pixel values from two images I and $\tilde{I}$, with $Y = \phi(X) + Z$, where Z is a real-valued white noise with probability density function $p(z)$ and independent of X. Using relations of the probabilities, we have

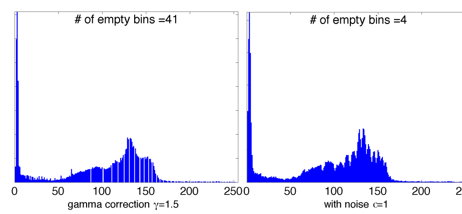

Fig. 3. *Effects of additive noise on the pixel histogram after contrast enhancement is applied.* (**Left**)*: pixel histogram of an image after gamma correction is applied,* (**Right**) *The same pixel histogram after white Gaussian noise of standard deviation* 1.0 *is added.*

$$\begin{aligned}
\Pr(Y = j) &= \textstyle\sum_i \Pr(Y = j | X = i) \Pr(X = i) \\
&= \textstyle\sum_i \Pr\left(Z \in [j - \phi(x) - \frac{1}{2}, j - \phi(x) + \frac{1}{2})\right) h_{i+1} \\
&= \textstyle\sum_{i,k} \Pr\left(Z \in [j - k - \frac{1}{2}, j - k + \frac{1}{2})\right) \mathbf{1}(\phi(i) = k) h_{i+1} \\
&= \textstyle\sum_{i,k} \Pr\left(Z \in [j - k - \frac{1}{2}, j - k + \frac{1}{2})\right) (T_\phi)_{i+1,k+1} h_{i+1} \\
&= \textstyle\sum_{i,k} \left(\int_{j-k-\frac{1}{2}}^{j-k+\frac{1}{2}} p(z) dz \right) (T_\phi)_{i+1,k+1} h_{i+1}
\end{aligned}$$

We introduce a new matrix

$$R_{j+1,k+1} = \int_{j-k-\frac{1}{2}}^{j-k+\frac{1}{2}} p(z) dz,$$

which has a Toeplitz structure and can be obtained in closed form for certain type of noise. For instance, if Z is zero-mean Gaussian noise with standard deviation σ, elements of matrix R are explicitly given as:

$$R_{j+1,k+1} = \operatorname{erf}\left(\frac{j - k + \frac{1}{2}}{\sigma}\right) - \operatorname{erf}\left(\frac{j - k - \frac{1}{2}}{\sigma}\right),$$

$\operatorname{erf}(z) = \int_{-\infty}^{u} \frac{1}{\sqrt{2\pi}} e^{-\frac{\tau^2}{2}} d\tau$ is the cumulative density function of the standardized Gaussian random variable. Subsequently, we assume R is known or can be obtained from images using methods of blind noise estimation, *e.g.*, [17, 24]. Considering the noise effect, the relation between the pixel histograms of the original image and the image after contrast enhancement is applied and white noise added can be expressed as

$$\tilde{\mathbf{h}} = R T_\phi \mathbf{h}, \tag{5}$$

which will be used subsequently to recover the contrast enhancement.

3.3 Estimating Contrast Enhancement from Pixel Histogram

The problem we are to solve is to recover the unknown contrast enhancement transform ϕ and the pixel histogram of the original image $\mathbf{h}$ simultaneously, using only the pixel histogram of the observed image $\tilde{\mathbf{h}}$.

We use the Wasserstein distance[21] (also known as Mallows distance or earth mover's distance[15] and subsequently referred to as W_1 distance) to measure distances between two histograms pixel histograms, $\mathbf{h}, \tilde{\mathbf{h}} \in \Delta^{n+1}$, defined as

$$W_1(\tilde{\mathbf{h}}, \mathbf{h}) = \sum_i \left|C_{Y\sim\tilde{\mathbf{h}}}(i) - C_{X\sim\mathbf{h}}(i)\right| = \left\|F\tilde{\mathbf{h}} - F\mathbf{h}\right\|_1, \tag{6}$$

where $C_{X\sim\mathbf{h}}(i) = \Pr(X \leq i) = \sum_{j=1}^{i+1} h_j$ is the cumulative distribution of $\mathbf{h}$. Note that $C_{X\sim\mathbf{h}}$ can be computed as $F\mathbf{h}$, where F is the lower triangular matrix with all elements equal to one.

Using the W_1 distance between two pixel histograms and the observation that contrast enhancement leads to non-smooth pixel histogram, we can formulate contrast enhancement estimation as the following optimization problem

$$\min_{\mathbf{h},\phi} W_1(\tilde{\mathbf{h}}, RT_\phi\mathbf{h}) + \lambda\Omega(\mathbf{h}), \text{ s.t. } \mathbf{h} \in \Delta^{n+1}, \phi(i) \leq \phi(i+1), i = 0, \cdots, n-1. \tag{7}$$

The first term in the objective function corresponds to relations of pixel histogram of image with and without contrast enhancement in Eq.(5). The second term reflects the observation in Section 3.1 that pixel histogram of the original image tend to have fewer number of empty bins. Parameter λ controls the contribution of the two terms in the objective function. The constraint enforces $\mathbf{h}$ to be a legitimate pixel histogram and ϕ to be a monotonic mapping.

We solve (7) using a block coordinate descent scheme [2] by alternating minimizing the objective with regards to $\mathbf{h}$ and ϕ with the other fixed. The estimation of $\mathbf{h}$ with fixed ϕ reduces to a convex optimization problem [3], and we describe its solution in Section 3.3.1. The estimation of ϕ with fixed $\mathbf{h}$ is then given in Section 3.3.2 for the parametric case and 3.3.3 for the nonparametric case.

3.3.1 Recovering Pixel Histogram with Known Contrast Enhancement. Using the equivalent definition of Wasserstein distance given in Eq.(6), the problem of finding optimal $\mathbf{h}$ with known ϕ is obtained from (7) as

$$\min_{\mathbf{h}} \left\|F\tilde{\mathbf{h}} - FRT_\phi\mathbf{h}\right\|_1 + \lambda\Omega(\mathbf{h}) \text{ s.t. } \mathbf{1}^T\mathbf{h} = 1, \mathbf{h} \geq 0. \tag{8}$$

Eq.(8) is difficult to solve because (i) the W_1 distance uses the non-differentiable ℓ_1 norm and (ii) $\Omega(\mathbf{h})$ is not a continuous function. To proceed, in this work, we replace the non-differentiable $\|\cdot\|_1$ using a generalization of the result in [18] (Theorem 2, proof in Appendix 6).

THEOREM 2. *For $\mathbf{x} \in \mathcal{R}^n$, we have*

$$\|\mathbf{x}\|_1 = \min_{\mathbf{z}\geq 0} \frac{1}{2}\left(\mathbf{x}^\top\mathcal{D}(\mathbf{z})^{-1}\mathbf{x} + \mathbf{1}^\top\mathbf{z}\right), \text{ and } |\mathbf{x}| = \operatorname*{argmin}_{\mathbf{z}\geq 0} \frac{1}{2}\left(\mathbf{x}^\top\mathcal{D}(\mathbf{z})^{-1}\mathbf{x} + \mathbf{1}^\top\mathbf{z}\right). \tag{9}$$

$\mathcal{D}(\mathbf{z})$ denotes a diagonal matrix formed from vector $\mathbf{z}$ as its main diagonal, and $|\mathbf{x}|$ as the vector formed from the absolute values of the components of $\mathbf{x}$.

Furthermore, note that for $\rho > 0$, we have $e^{-\rho c} \geq \mathbf{1}_{c=0}$ with equality holding iff $c = 0$, and for $c > 0$, $e^{-\rho c} \to 0$ with $\rho \to \infty$. This means that we can use $e^{-\rho\mathbf{h}}$, where the exponential is applied to each element of vector $\mathbf{h}$, as a continuous and convex surrogate to the non-continuous function in $\Omega(\mathbf{h})$.

Using these results, we can develop an efficient numerical algorithm. First, we introduce an auxiliary variable $\mathbf{u} \geq 0$ to replace the ℓ_1 norms in (8), and use the scaled exponential function to reformulate the problem as

$$\min_{\mathbf{h},\mathbf{u}} L(\mathbf{h}, \mathbf{u}), \text{ s.t. } \mathbf{1}^T\mathbf{h} = 1, \mathbf{h} \geq 0, \mathbf{u} \geq 0, \tag{10}$$

with

$$L(\mathbf{h}, \mathbf{u}) = \frac{1}{2}\left(\tilde{\mathbf{h}} - RT_\phi\mathbf{h}\right)^\top F^\top\mathcal{D}(\mathbf{u})^{-1}F\left(\tilde{\mathbf{h}} - RT_\phi\mathbf{h}\right) + \frac{1}{2}\mathbf{1}^\top\mathbf{u} + \lambda e^{-\rho\mathbf{h}}.$$

This is a constrained convex optimization problem for $(\mathbf{h}, \mathbf{u})$. While in principle we can use off-the-shelf convex programming packages such as CVX [8] to solve this problem, the potentially high dimensionality of $\mathbf{h}$ (for instance for 14-bit image $\mathbf{h}$ is of 16, 384 dimensions) requires a more efficient algorithm designed for our problem. Our algorithm is provided in the pseudo-code in Algorithm 1 and the details of the derivation of this algorithm can be found in the Appendix.

ALGORITHM 1: Optimization of (8)

```
initialize h_0 and u_0, t ← 0
while not converge do
    h_{t,0} ← h_t, τ ← 0
    while not converge do
        M ← (FRT_φ)^⊤ D(u_t)^{-1} (FRT_φ)
        b ← (FRT_φ)^⊤ D(u_t)^{-1} F h̃
        △h_τ ← M h_{t,τ} − b − λρ e^{−ρ h_{t,τ}} h_{t,τ}
        h_{t,τ+1} ← P_{Δ^{n+1}} (h_{t,τ} − η_0/(τ+1) △h_τ)
        τ ← τ + 1
    end while
    h_{t+1} ← h_{t,τ}, u_{t+1} ← |F (h̃ − RT_φ h_{t+1})|
    t ← t + 1
end while
```

3.3.2 *Estimating Parametric Contrast Enhancement.* For a parametric contrast enhancement transform that can be determined by small set of parameters θ, *e.g.*, gamma correction (1) (where $\theta = \gamma$) and sigmoid stretching (2) (where $\theta = (\alpha, \mu)$), even though the parameters are continuous, the discrete nature of contrast enhancement as transforms between integers means that there are only finite number of distinguishable parameter values. We illustrate this point in the case of gamma correction. Consider the 2D lattice $\{0, \cdots, n\} \times \{0, \cdots, n\}$, the graph of gamma correction corresponds to a path over grid points (i, j) starting from $(0, 0)$ and ending at (n, n). This path is monotonic, *i.e.*, it never dips down. Furthermore, for $\gamma < 1$, the path stays on or above the diagonal, while for $\gamma > 1$, the path stays on or below the diagonal. These properties give rise to only a finite set of distinguishable γ values, (*i.e.*, values leading to different gamma correction transforms) as the following result shows (proved in Appendix).

THEOREM 3. *All* $\gamma \in [\underline{\gamma}_{ij}, \bar{\gamma}_{ij})$ *for* $i, j \in \{1, \cdots, n-1\}$ *where*

$$\underline{\gamma}_{ij} = \frac{\log n - \log(j + \frac{1}{2})}{\log n - \log i}, \bar{\gamma}_{ij} = \frac{\log n - \log(j - \frac{1}{2})}{\log n - \log i}.$$

leads to the same gamma correction curve. As such, the total number of distinguishable γ *value is bounded by* $(n-1)^2$.

In practice, distinguishable parameter values are also limited by the numerical precision in which they can be input in photo editing software, usually in the range of 10^{-2} or 10^{-3}.

ALGORITHM 2: Estimation of Parametric Contrast Enhancement

```
for θ ∈ {θ_1, ··· , θ_m} do
    compute L*(θ) using Algorithm 1;
end for
return θ* = argmin_{θ∈{θ_1,···,θ_m}} L*(θ)
```

On the other hand, optimal contrast enhancement parameters lead to a minimum of Eq.(10) across different parameter values. As the transformed pixel histogram will be exactly the same as the observed histogram, thus the first term will reach minimum (zero), while the original histogram should have the minimum number of zero bins. We denote the minimum of (10) corresponding to contrast enhancement parameter θ as $L^\star(\theta)$. These two characteristics of parametric contrast enhancement, *i.e.*, the finite number of distinguishable parameter values and the optimal value leading to the global minimum of (10), suggest that the optimal parameter can be recovered by a grid search in the set of plausible parameters. Specifically, given a search range of parameter values $\Phi = \{\theta_1, \cdots, \theta_m\}$, we seek $\theta^\star = \text{argmin}_{\theta \in \Phi} L^\star(\theta)$ as the optimal contrast enhancement parameter. This is the algorithm we use for estimating parametric contrast enhancement. A pseudo code is given in Algorithm 2.

3.3.3 *Estimating Nonparametric Contrast Enhancement.* In the case where the contrast enhancement transform does not afford a parametric form, with the pixel histogram of the original image obtained using the algorithm given §3.3.2, we estimate ϕ directly. Dropping irrelevant terms from the overall optimization problem (7), this reduces to the following problem

$$\min_{\phi} W_1(\tilde{\mathbf{h}}, RT_\phi \mathbf{h}) \text{ s.t. } \phi(i) \leq \phi(i+1), i = 0, \cdots, n. \quad (11)$$

To solve this problem, we first decouple pixel histogram before and after noise is added. To this end, we introduce an auxiliary variable $\hat{\mathbf{h}}$ and a parameter ξ to enforce the constraint $\hat{\mathbf{h}} = T_\phi \mathbf{h}$ with W_1 distance, and rewrite the optimization problem as

$$\min_{\hat{\mathbf{h}}, \phi} W_1(\tilde{\mathbf{h}}, R\hat{\mathbf{h}}) + \xi W_1(\hat{\mathbf{h}}, T_\phi \mathbf{h}) \text{ s.t. } \hat{\mathbf{h}} \in \Delta^{n+1}, \phi(i) \leq \phi(i+1), i = 0, \cdots, n. \quad (12)$$

Using the block coordinate descent scheme, we solve (14) by alternating minimization of $\hat{\mathbf{h}}$ and ϕ with the other fixed until the guaranteed convergence is reached.

Optimizing $\hat{\mathbf{h}}$ This problem becomes

$$\min_{\hat{\mathbf{h}}, \phi} \left\| F\tilde{\mathbf{h}} - FR\hat{\mathbf{h}} \right\|_1 + \xi \left\| F\hat{\mathbf{h}} - FT_\phi \mathbf{h} \right\|_1 \text{ s.t. } \mathbf{1}^T \hat{\mathbf{h}} = 1, \hat{\mathbf{h}} \geq 0. \quad (13)$$

Following a similar procedure as for the solution of Eq.(8), we optimize (13) with an iterative algorithm that uses the ℓ_1 relaxation in Theorem 2 and projected gradient descent. We defer a detailed derivation and algorithm to the Appendix.

Optimizing ϕ. We use the equivalent definition of W_1 distance based on the cumulative distributions (6), and the problem of solving ϕ reduces to

$$\min_{\phi} \sum_{i=0}^{n} \left| C_{\hat{X} \sim \hat{\mathbf{h}}}(i) - C_{\phi(X) \sim T_\phi \mathbf{h}}(i) \right| \text{ s.t. } \phi(i) \leq \phi(i+1), i = 0, \cdots, n. \quad (14)$$

This is essentially the search for a monotonic transform between two random variables X and $\hat{X}$ over $\{0, \cdots, n\}$ with corresponding probability distributions (histograms) $\mathbf{h}$ and $\hat{\mathbf{h}}$, such that the histogram of $\phi(X)$ is as close as possible to that of $\hat{X}$.

ALGORITHM 3: Estimation of Nonparametric Contrast Enhancement

```
while not converge do
    update h̃ (8) using Algorithm 1;
    update ĥ (13) using Algorithm 4;
    update φ (14) with (16);
end while
```

The problem of finding a transform that matches random variable of one distribution to another is known as *histogram matching*, the optimal solution of which can be obtained from the cumulative distributions of the two random variables [21]. Specifically, from the cumulative probability distribution of $\mathbf{h}$, $C_{X \sim \mathbf{h}} : \{0, \cdots, n\} \mapsto [0, 1]$, we define the corresponding pseudo inverse cumulative distribution function $[0, 1] \mapsto \{0, \cdots, n\}$, as

$$C^{-1}_{X \sim \mathbf{h}}(j) = i \text{ if } C_{X \sim \mathbf{h}}(i-1) < j \leq C_{X \sim \mathbf{h}}(i). \quad (15)$$

The histogram matching transform is formed by applying the cumulative distribution function of $\mathbf{h}$ followed by the pseudo inverse cumulative distribution function of $\tilde{\mathbf{h}}$ (15),

$$\phi^\star(i) = C^{-1}_{Y \sim \tilde{\mathbf{h}}}(C_{X \sim \mathbf{h}}(i)). \quad (16)$$

It can be shown that this function is monotonic and maps $X \sim \mathbf{h}$ to $\phi^\star(X) \sim \tilde{\mathbf{h}}$ and leads to the objective function in (14) to zero. We provide the pseudo code of the overall algorithm in Algorithm 3.

4 EXPERIMENTAL EVALUATION

In this section, we report experimental evaluations of the contrast enhancement estimation method described in the previous section. The images used in our experiments are based on $N = 2,000$ grayscale images from the RAISE dataset[9]. The original images are in the 12-bit or 14-bit uncompressed or lossless compressed NEF or TIFF format. We downloaded the full RAISE dataset but use a random subset of 2,000 images that are further cropped to the same size of 1000×800. We use the green channel of the RGB color image as in [6]. The pixel histograms are vectors of $2^{12} = 4,096$ and $2^{14} = 16,384$ dimensions, respectively. All subsequent running time statistics are based on a machine of 3.2GHz Due Core Intel CPU and 16G RAM and unoptimized MATLAB code.

4.1 Parametric Contrast Enhancement Estimation

4.1.1 Gamma Correction. We first consider the estimation of gamma correction transform. We choose γ value from the range of $\{0.1 : 0.05 : 2.5\}$, with each γ value applied to all N images to create sets of gamma corrected images. We implement the grid-search based algorithm (Algorithm 2 in §3.3.2) using a probing range of $\{0.1 : 0.01 : 2.5\}$ to recover the γ values from these images. We use a stride of 0.01 as it is the minimum numerical precision a user can specify a gamma correction in image editing tools. We choose parameter $\lambda = 0.75$ and $\rho = 1$ as we found the estimation results are not particularly sensitive to the values of these parameters. Unless specified, we set the noise level to $\sigma = 0.01$. Fig.4 shows the graph of $L^\star(\theta)$ for the case of gamma correction. The curves correspond to two different parameter values: $\gamma = 0.4$ and 1.4, and the range of the searched γ values is set to $[0.1, 1.8]$ with a step size of 0.05. In both cases, the true γ values lead to global minimums of $L^\star(\theta)$[2].

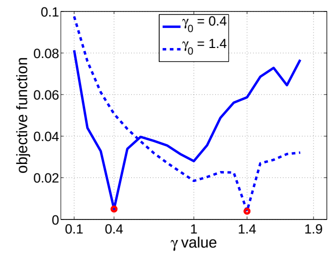

Fig. 4. *The relation between the value of $L^\star(\gamma, \lambda)$ and γ values, in the case of contrast mapping in the form of simple gamma correction. The true γ value in the two cases are 0.4 and 1.4, corresponding to the global minima.*

We use the estimation accuracy rate (AR) to quantify the estimation performance. For an error threshold ϵ, A_ϵ corresponds to the fraction of estimations that are within a relative error of ϵ. Specifically, denoting the true parameter as $\gamma^\star$ and the estimated parameter as γ_i for each of the N test images, AR is defined as

$$A_\epsilon = \frac{\sum_{i=1}^{N} \mathbf{1}(|\gamma_i - \gamma^\star| \leq \epsilon)}{N}. \tag{17}$$

For a given ϵ, higher AR A_ϵ corresponds to better estimation performance. Subsequently, we report A_0, $A_{0.01}$ and $A_{0.05}$, corresponding to the cases when the estimation is exact, has a relative error ≤ 0.01 and has a relative error ≤ 0.05, respectively.

We apply our estimation algorithm and compare it with two previous works on gamma correction estimation [6, 10][3]. The results for A_0, $A_{0.01}$ and $A_{0.05}$ for the full range of probing γ values are shown in Fig.5. The bi-spectra based method of [10] demonstrates some stable estimation results for γ value near 1.0, yet the performance deteriorates as γ deviates from 1.0. This may be due to the fact that estimations of bi-spectral features become less reliable for more extreme γ values. The original method of [6] is a classification scheme based on the empty-bin locations as classification features. To apply it to the estimation problem, we build 250 classifiers corresponding to the probing range of γ values, and output the γ value that corresponds to the largest classification score. Using only the locations of the empty bins may not be sufficient to recover the γ value as many neighboring γ values share similar empty bin locations. On the other hand, our method achieves significant improvement in performance when comparing with those of the two previous works. We attribute the improved estimation performance to that the optimization formulation of the problem better captures characteristics of pixel histogram, with the Wasserstein loss reflects different locations

[2]Similar observations have also been made on other types of parametric contrast enhancement transforms, such as sigmoid stretching and cubic spline curves.

[3]We use our own MATLAB implementation of these methods following the settings provided in the corresponding published papers. We compared only with the results of [6] which improves on the earlier work from the same authors in [5].

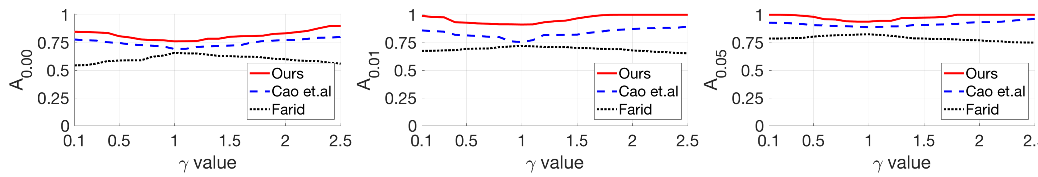

Fig. 5. *Performance comparison of parametric contrast enhancement estimation methods for gamma correction, the three plots corresponds to A_0, $A_{0.01}$ and $A_{0.05}$, respectively.*

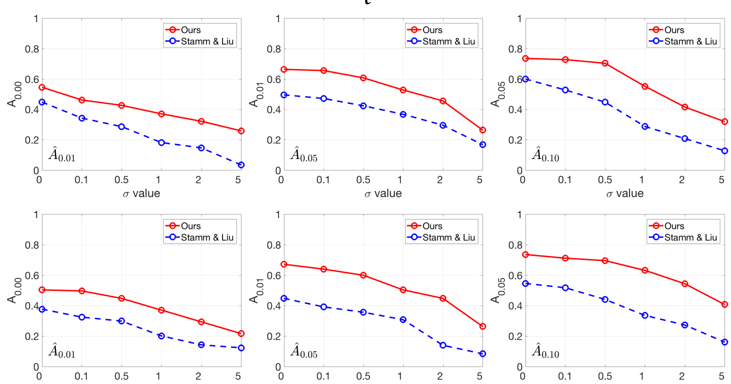

Fig. 6. *Performance comparison of nonparametric contrast enhancement estimation methods for histogram equalization (top) and the freeform contrast enhancement curve (bottom), the three rows corresponding to plots of $\hat{A}_{0.01}$, $\hat{A}_{0.05}$ and $\hat{A}_{0.10}$, respectively.*

of the empty bin, and the regularizer favoring smaller number of empty bins further reduces uncertainty in determining the γ value. The average running time is 23.1 second per test image of size 1000×800 pixels, as the algorithm iterates over 250 different gamma values.

actual α/μ	estimated α (mean/std)	estimated μ (mean/std)
0.5 / -1.0	0.49 (0.01)	-1.02 (0.01)
0.5 / 0.0	0.50 (0.02)	0.01 (0.01)
0.5 / 1.0	0.50 (0.01)	1.01 (0.02)
1.0 / -1.0	0.98 (0.02)	-0.99 (0.01)
1.0 / 0.0	1.02 (0.04)	0.02 (0.03)
1.0 / 1.0	1.01 (0.02)	1.04 (0.03)
2.0 / -1.0	2.02 (0.03)	-0.98 (0.02)
2.0 / 0.0	2.01 (0.03)	0.99 (0.03)
2.0 / 1.0	1.97 (0.02)	0.98 (0.01)

Table 1. *Performances of estimating sigmoid stretching on* 100 *test images with different* (α, μ) *values.*

4.1.2 Sigmoid Stretching. The next experiment tests the performance of our method on the estimation of parameters of sigmoid stretching, Eq.(2). To our best knowledge, there is no previous method developed for the estimation of this parametric contrast enhancement transform. To this end, we created test images using the range of parameters $(\alpha, \mu) \in \{0.5, 1.0, 2.0\} \times \{-1.0, 0.0, 1.0\}$, and our algorithm performs a grid search in the range of $\{0.2 : 0.01 : 2.5\} \times \{-1.5 : 0.01 : 1.5\}$. The results, as the averages and standard deviations of the estimated (α, μ) values over the 100 test image, are listed in Table 1. Our method is effective to recover the original contrast enhancement parameters. However, the two dimensional search space increase the running time to 15 seconds per image.

4.1.3 Robustness under Noise and JPEG. We further evaluate the robustness of our method in the presence of noise and JPEG compression. We apply gamma correction with γ values randomly sampled from the range $[0.1, 2.5]$ to generate $2,000$ test images. Then, white Gaussian noises with zero mean and various levels are added to the gamma corrected image and then rounded to integral pixel values. We further applied JPEG compression of different quality factors to the same set of gamma corrected images.

Shown in Fig 7 are the performance evaluated with $A_{0.05}$, which is the percentage of estimated γ that fall in the range of ± 0.05 of the ground truths, as well as comparisons with the methods of [10] and [6]. Accuracies of all methods are affected by the additive noises and JPEG compression. But in the case of noise, the performances of our method show less degradation in comparison with those of the previous works because our method directly incorporate noise perturbations, while the previous works are based on properties that are fragile in the presence of perturbations. On the other hand, in the presence of JPEG compression, our method achieves comparable performance with the method of [6] that is specifically designed to model the artifacts introduced by JPEG to an image after contrast enhancement.

4.2 Nonparametric Contrast Enhancement

We further test our methods to recover two different types of nonparametric contrast enhancement transforms – histogram equalization and a nonparametric contrast enhancement created by cubic spline interpolation of manually selecting key points. The latter is analogous to free-form contrast enhancement transform in photo-editing software (*e.g.*, the `Curve` tool in `Photoshop`). We applied histogram equalization and interpolated contrast enhancement transform to create 100 test images of each type.

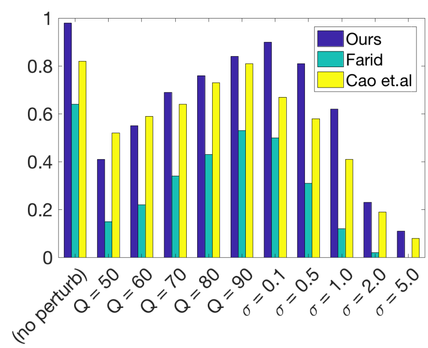

Fig. 7. *Robustness under additive Gaussian noise and JPEG compression for the estimation of gamma correction evaluated by $A_{0.05}$.*

Because of the nonparametric nature of the contrast enhancement transform, we measure the performance using a slightly different metric based on the relative root mean squared error (RMSE) between $\phi^\star$ and ϕ_k to evaluate the performance. Denote the true contrast enhancement transform as $\phi^\star$ and the estimated contrast enhancement transform using each test image as ϕ_k, for an error threshold ϵ, we define the accuracy rate as $\hat{A}_\epsilon = \frac{1}{N}\sum_{k=1}^{N} \mathbf{1}\left(\frac{\|\phi_k-\phi^\star\|_2}{\|\phi^\star\|_2} \le \epsilon\right)$.

We implemented our algorithm to recover nonparametric contrast enhancement as described in Algorithm 3 in Section 3.3.3, and set $\lambda = 0.75$ and $\xi = 10$. In practice, we observe that the algorithm usually converges within less than 10 iterations. Fig.8 demonstrate the convergence of one estimated contrast enhancement transform, with the original transform obtained from interpolating manually chosen key points using cubic splines. As it shows, after 5 iterations of the algorithm, the estimated transform is already very close to the true transform.

We compare it with the only known previous work for the same task in [19], which is based on an iterative and exhaustive search of pixel histogram that can result in the observed pixel histogram of an image after the contrast enhancement is applied[4].

Fig.6 shows the performance of both algorithms measured by $\hat{A}_{0.01}$, $\hat{A}_{0.05}$ and $\hat{A}_{0.10}$, with perturbations from additive white Gaussian

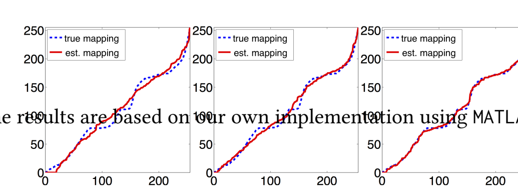

Fig. 8. *Convergence of the estimated nonparametric free-form contrast enhancement transform.*

[4]Code of this work is not made public by the authors, so the results are based on our own implementation using MATLAB following the descriptions in [19].

noise of different strengths. As these results show, the estimation performance with our method is consistently better than that of [19]. Furthermore, our method takes about 2.2 seconds to run on an 1000×800 image and on average it is $5-10$ times faster than the method of [19], which relies on an exhaustive search. More importantly, the performances of the method in [19] seem to be strongly affected by noise and compression, this is in direct contrast to our method, which can take such perturbation into consideration to become more robust.

5 LOCAL CONTRAST ENHANCEMENT ESTIMATION

A composite image can be created by replacing a region in one image with a region from a different image. If the host and the donor images were captured under different lighting environments, an image forger can perform contrast enhancement so that lighting conditions match across the composite image. Image forgeries created in this manner can thus be revealed with inconsistent contrast enhancement transforms across different regions using a local contrast enhancement estimation method.

A straightforward approach would be to apply the global contrast enhancement detection method to non-overlapping rectangular blocks of pixels [6, 16, 19]. However, this simple method has several problems. First, due to the smaller number of pixels in each block, it is difficult to obtain a reliable estimation of the contrast enhancement transform. Second, this simple approach does not take into consideration that adjacent blocks are likely to have undergone the same contrast enhancement. The third problem with these methods is that, to avoid long running time, these methods are only run on non-overlapping blocks and obtain *block level* localization, while for practical forensic analysis it is desirable to have *pixel level* localization of regions undergone different contrast enhancement. To improve on these aspects, in this section, we describe a new local contrast enhancement estimation method based on our global estimation method, but embed it in an energy minimization formulation for a more effective and efficient local contrast enhancement estimation at the pixel level.

5.1 Energy Minimization

We segment an image into m overlapping blocks, $\{I_1, \cdots, I_m\}$, and denote $\mathscr{N}(k) \subseteq \{1, \cdots, m\}$ as the indices of blocks that are spatial neighbors of block I_k based on a 4-connected neighborhood system. We use an operator $\mathcal{H}(I)$ to denote the procedure of obtaining pixel histogram from an image region I.

We assume that there are two regions in the image undergone two different and unknown contrast enhancement transforms, ϕ_0 and ϕ_1, and associate each block with a binary label $y_k \in \{0, 1\}$: $y_k = 0$ indicating that ϕ_0 is applied to I_k and $y_k = 1$ indicating I_k has contrast enhancement ϕ_1[5]. Our algorithm obtains estimation of ϕ_0 and ϕ_1 and image regions to which they are applied simultaneously, which is formulated as minimizing the following energy function with regards to ϕ_0, ϕ_1 and labels $\{y_k\}_{k=1}^m$:

$$\sum_k \mathcal{E}_{y_k}(I_k) + \beta \sum_k \sum_{k' \in \mathcal{N}(k)} |y_k - y_{k'}|. \tag{18}$$

[5]The subsequent algorithm can be readily extended to the case of more than two regions by replacing the binary label with multi-valued labels.

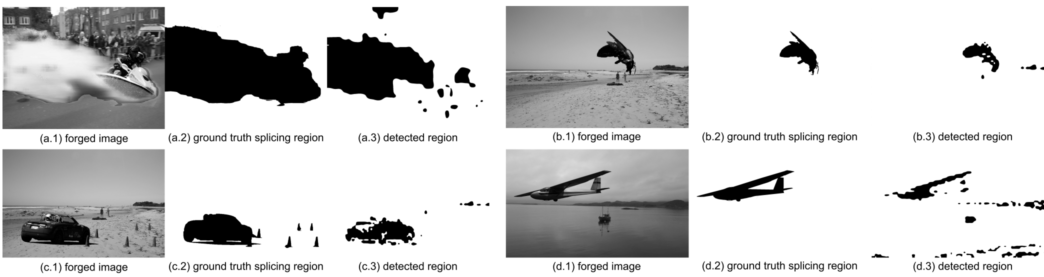

Fig. 9. *Local contrast enhancement estimation for tampering detection.* **(a1, b1, c1, d1)***: Four examples of the manipulated images created from the NIMBLE dataset.* **(a2, b2, c2, d2)***: Ground truth masks of the spliced regions.* **(a3,b3,c3,d3)***): The detection results with black and white corresponding to regions with label* 0 *and* 1.

The first term in Eq.(18) is the unary energy or data term for the two values of the label, which is defined as

$$\begin{aligned} \mathcal{E}_0(I) &= \min_{\mathbf{h}\in\Delta^{n+1}} \log\left(W_1(\mathcal{H}(I), RT_{\phi_0}\mathbf{h}) + \lambda\Omega(\mathbf{h})\right) \\ \mathcal{E}_1(I) &= \min_{\mathbf{h}\in\Delta^{n+1}} \log\left(W_1(\mathcal{H}(I), RT_{\phi_1}\mathbf{h}) + \lambda\Omega(\mathbf{h})\right). \end{aligned} \tag{19}$$

Note that the unary energy is obtained as the result of optimizing Eq.(8) assuming known contrast enhancement, thus it can be computed with Algorithm 1.

The second term in Eq.(18), $\beta \sum_k \sum_{k'\in\mathcal{N}(k)} |y_k - y_{k'}|$, corresponds to the binary energy that penalizes differences of label assignments to neighboring blocks. It reflects the assumption that the same contrast enhancement is applied to an extended region in the image that subsumes many neighboring blocks. Parameter β is used to balance the numerical contribution of the unary and binary energy in the overall energy function (we use $\beta = 0.1$ in the subsequent experiments).

The minimization of Eq.(18) is a mixed optimization problem with discrete labels and continuous functions, and we solve it by an iterative block coordinate descent algorithm that alternates between the optimization of (ϕ_0, ϕ_1) and $\{y_k\}_{k=1}^m$ with the other set of variable fixed.

Optimizing $\{y_k\}_{k=1}^m$. With fixed ϕ_0 and ϕ_1, the unary energy in Eq.(19) can be computed with Algorithm 1. The energy function is a sub-modular function of the binary labels $\{y_k\}_{k=1}^m$, which can be minimized using the graph cut algorithm [4].

Optimizing (ϕ_0, ϕ_1). The update for ϕ_0 and ϕ_1 with fixed cluster labels proceeds as re-estimating ϕ_0 and ϕ_1 using the union of all pixels in blocks with the corresponding label equal to 0 and 1, respectively, as

$$\begin{aligned} \phi_0 &\leftarrow \operatorname{argmin}_\phi \min_{\mathbf{h}} W_1(\mathcal{H}(\cup_{k:y_k=0} I_k), RT_{\phi_0}\mathbf{h}) + \lambda\Omega(\mathbf{h}) \\ \phi_1 &\leftarrow \operatorname{argmin}_\phi \min_{\mathbf{h}} W_1(\mathcal{H}(\cup_{k:y_k=1} I_k), RT_{\phi_0}\mathbf{h}) + \lambda\Omega(\mathbf{h}) \end{aligned} \tag{20}$$

This is implemented as first collecting pixel histogram on these blocks, then apply Algorithm 4 in Section 3.3.3 to recover the contrast enhancement transform[6]. Compared to previous methods that are based on single image blocks, this increases the number of pixels in the estimation of pixel histogram and thus improves the stability of the estimation.

5.2 Experimental Evaluation

We perform experimental evaluations of the local contrast enhancement estimation algorithm using a set of 500 composite images with ground truth masks of the spliced regions. These images are a subset of the NIMBLE Challenge dataset provided by NIST for evaluating existing image forensic methods[7]. These images were generated by composing different regions from donor images and pasted into the tampered image. The typical size of the tampered region is about 15 − 35% of the size

[6]We did not use the grid-search based Algorithm 3 to handle the parametric contrast enhancement as in practical scenarios we usually do not have the knowledge of the types of contrast enhancement involved.
[7]These images can be downloaded from https://mig.nist.gov/MediforBP/MediforBPResources.html.

of the original images. We applied different contrast enhancement transforms to the spliced regions. Because the masks of the spliced regions of these composite images are provided, we generate locally contrast enhancement transformed images by using taking the tampered region then applied contrast enhancement operations to the tampered regions. The NIMBLE images are in JPEG format and to avoid introducing double-JPEG artifact, we compress the resulting image with the same JPEG qualities. To ensure diversity of the applied contrast enhancement, we choose from four different cases, including gamma correction, sigmoid stretching, histogram equalization and monotonic cubic spline curve with hand picked control points. Four examples of the manipulated images are shown in the left column of Fig.9, with the ground truth masks of spliced region shown in the middle column.

We implement the local contrast enhancement estimation algorithm described in the previous section. We try to simulate a situation where the specific form of contrast enhancement is unknown, so we use Algorithm 4 in Section 3.3.3 and ignore the fact that some of the contrast enhancements have parametric form. The size and stride of these blocks determine the reliability of the estimated contrast enhancement transforms and the accuracy in locating the spliced regions in a composite image. Empirically, we found block size of 50×50 pixels with overlapping strides of 2 pixels to provide a good tradeoff of running efficiency and estimation accuracy, so they are used throughout our subsequent experiments.

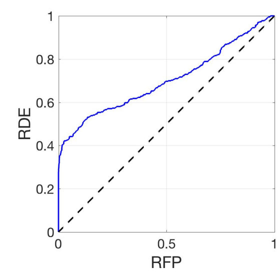

Fig. 10. *ROC curve of the spliced region detection based on the contrast enhancement detection method. The Area under ROC (AUC) is* 72.5%.

The detection results on the four examples given in the leftmost column of Fig.9 are shown in the corresponding panels on the right, with black and white corresponding to regions with label 0 and 1. To quantitatively evaluate the results, we use the region detection ratio (*RDE*) and region false positive ratio (*RFP*) to measure the accuracy of the recovered region undergone the same contrast enhancement transform. Specifically, with R_D and R_T corresponding to the detected and true region undergone ϕ_0, respectively, with $|R|$ representing the area of an image region R, we define $DE = |R_T \cap R_D|/|R_T|, FP = 1 - |R_T \cap R_D|/|R_D|$. Note that these two rates vary as the threshold that we use to generate the binary mask. To evaluate the performance, we show the ROC curve in Fig.10 averaged over the 500 images from the NIMBLE dataset. The Area under ROC of this plot is 72.5%. On the whole image level, for all 500 images there are more than 50% of the spliced region detected. As these results show, our method is capable of recovering the majority of the spliced regions. The averaged running time over a $2,000 \times 1,500$ image is about 56.2 seconds on a machine of 3.2GHz Due Core Intel CPU and 16G RAM and MATLAB implementation of the algorithm.

On the other hand, we also noticed that large continuous regions with few pixel values due to large monotone or insufficient exposure can lead to false positive or mis-detections. The pixel histogram of these areas are usually sparse and a contrast enhancement transform just move bins around without changing the number of empty bins significantly. A future work is to identify such regions based on their pixel histogram and exclude them from the estimation procedure.

6 CONCLUSION

In this work, we describe a new method to estimate contrast enhancement from images, taking advantage of the nature of contrast enhancement as a mapping between integral pixel values and the distinct characteristics it introduces to the pixel histogram of the transformed image. Our method recovers the original pixel histogram and the applied contrast enhancement simultaneously with an efficient iterative algorithm, and can effectively handle perturbations due to noise and compression. We perform experimental evaluation to demonstrate the efficacy and efficiency of the proposed method. By examining local areas in the image, we also show that using this method, we can detect spliced image regions that have undergone different contrast enhancement transformations.

There are several shortcomings to the current method that provide directions in which we would like to further improve the current work in the future. First, we can further improve the robustness of the current algorithm robust with regards to JPEG compressions with different quality factors, and we are investigating modeling JPEG compression using similar mathematical framework and handling them as in [6, 12]. Second, the current localization algorithm cannot effectively handle large image regions with monotone content or that are over or under exposure. Such areas lead to unstable estimations and should be opt-outed from the analysis. Third, our current method is based on the assumption that contrast enhancement operations are among the last manipulations performed on the image or image regions, such that the traces they leave in the pixel histogram are still significant to be exposed. This is certainly a strong assumption that may not hold in actual image manipulations. Therefore, we will further improve the global and local estimation method to efficiently work in a real forensic contexts where the quality factors and processing orders are unknown *a priori*.

Acknowledgement. This work is partially supported by the US DARPA Media Forensic (MediFor) Program and National Science Foundation of China Research Grant (#61771341). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the DARPA and NSF.

APPENDICES

Proof of Theorem 1. We consider a non-empty bin in the pixel histogram of an image. As the contrast enhancement transform transports each bin as a whole (*i.e.*, no splitting of bins), there are only two situation can occur: either this bin becomes another individual non-empty bin in the contrast enhancement transformed image, or it is mapped to a location other bins are also mapped to. In either case, the number of non-empty bins will not increase, and correspondingly the number of empty bins in the pixel histogram after contrast enhancement is applied does not decrease. On the other hand, the strict inequality does not hold, as if we have a monotone image with a single distinct pixel value, any contrast enhancement will only create another monotone image, in which case the number of empty bins remains the same.

Proof of Theorem 2. First, for $x \in \mathcal{R}$, we have $|x| = \min_{z\geq 0} \frac{1}{2}\left(\frac{x^2}{z} + z\right) = \text{argmin}_{z\geq 0} \frac{1}{2}\left(\frac{x^2}{z} + z\right)$, since for $x \neq 0$, differentiating the objective with regards to z and setting results to zero give $\frac{1}{2}\left(-\frac{x^2}{z^2} + 1\right) = 0 \Rightarrow z = |x|$, and for $x = 0$ the obvious optimal solution is $z = 0$. Using this result, we have

$$\|\mathbf{x}\|_1 = \sum_{i=1}^n \min_{z_i \geq 0} \frac{1}{2}\left(\frac{x_i^2}{z_i} + z_i\right) = \min_{z_i \geq 0} \sum_{i=1}^n \frac{1}{2}\left(\frac{x_i^2}{z_i} + z_i\right) = \min_{\mathbf{z}\geq 0} \frac{1}{2}\left(\mathbf{x}^\top \mathcal{D}(\mathbf{z})^{-1}\mathbf{x} + \mathbf{1}^\top \mathbf{z}\right).$$

And so does the argmin part of the result.

Derivation of Algorithm 1. We solve (10) with a block coordinate descent sub-procedure by iterating $\mathbf{h}_{t+1} \leftarrow \text{argmax}_{\mathbf{h}} L(\mathbf{h}, \mathbf{u}_t)$ and $\mathbf{u}_{t+1} \leftarrow \text{argmax}_{\mathbf{u}} L(\mathbf{h}_{t+1}, \mathbf{u})$ until convergence. The overall algorithm minimizing (8) is summarized in pseudocode in Algorithm 1.

Optimizing h: fixing $\mathbf{u}_t$ and dropping irrelevant terms, minimizing $\mathbf{h}$ reduces to the following constrained nonlinear convex optimization problem

$$\min_{\mathbf{h}} \frac{1}{2}\mathbf{h}^\top M\mathbf{h} - \mathbf{b}^\top \mathbf{h} + \lambda e^{-\rho \mathbf{h}} \text{ s.t. } \mathbf{1}^T\mathbf{h} = 1, \mathbf{h} \geq 0, \tag{21}$$

where

$$M = \left(FRT_\phi\right)^\top \mathcal{D}(\mathbf{u}_t)^{-1}\left(FRT_\phi\right) \ \mathbf{b} \ = \left(FRT_\phi\right)^\top \mathcal{D}(\mathbf{u}_t)^{-1}F\tilde{\mathbf{h}}.$$

Eq.(21) can be efficiently solved with a projected gradient descent method [3]. Specifically, starting with $\mathbf{h}_{t,0} = \mathbf{h}_t$, our algorithm iterates between two steps:

$$\begin{array}{rl} \triangle\mathbf{h}_\tau & \leftarrow M\mathbf{h}_{t,\tau} - \mathbf{b} - \lambda\rho e^{-\rho\mathbf{h}_{t,\tau}}\mathbf{h}_{t,\tau}, \\ \mathbf{h}_{t,\tau+1} & \leftarrow \mathcal{P}_{\Delta^{n+1}}\left(\mathbf{h}_{t,\tau} - \frac{\eta_0}{\tau+1}\triangle\mathbf{h}_\tau\right). \end{array} \tag{22}$$

The first step computes the gradient of the objective function. The second equation performs a gradient descent update with step size $\frac{\eta_0}{\tau+1}$ followed by a projection onto the n-dimensional probability simplex $\mathcal{P}_{\Delta^{n+1}}$, which does not have closed-form but affords a very efficient algorithm (see Appendix for a detailed description for completeness). The damping step size $\frac{\eta_0}{\tau+1}$ guarantees the convergence of the projected gradient descent algorithm [3], and we choose $\eta_0 = 1.2$ in all our subsequent experiments. This projected gradient descent algorithm usually converges within 5-10 steps. We take $\mathbf{h}_{t+1} = \mathbf{h}_{t,\tau}$ at the convergence.

Optimizing u: when fixing $\mathbf{h}_{t+1}$ and dropping irrelevant terms, minimizing $\mathbf{u}$ becomes

$$\min_{\mathbf{u}} \mathbf{c}^\top\mathcal{D}(\mathbf{u})^{-1}\mathbf{c} + \mathbf{1}^\top\mathbf{u} \text{ s.t. } \mathbf{u} \geq 0,$$

where $\mathbf{c} = F\left(\tilde{\mathbf{h}} - RT_\phi\mathbf{h}_{t+1}\right)$. Using Theorem 2, we obtain the optimal solution $\mathbf{u}_{t+1} = |\mathbf{c}|$.

Projection on probability simplex. The projection of a vector $\mathbf{x}$ on Δ^{n+1} is defined as the solution to the following optimization problem

$$\min_{\mathbf{h}} \frac{1}{2}\|\mathbf{x}-\mathbf{h}\|^2 \text{ s.t. } \mathbf{h} \geq 0, \mathbf{1}^\top\mathbf{h} = 1. \tag{23}$$

Introducing Lagrangian multipliers $\mathbf{y} \geq 0$ and ξ, we form the Lagrangian of Eq.(23) as

$$\mathcal{L}(\mathbf{h},\mathbf{y},\xi) = \frac{1}{2}\|\mathbf{x}-\mathbf{h}\|^2 - \mathbf{y}^\top\mathbf{h} - \xi(\mathbf{1}^\top\mathbf{h} - 1).$$

The corresponding KKT condition is then given by

$$\begin{array}{lr} \mathbf{h} - \mathbf{x} - \mathbf{y} - \xi\mathbf{1} = 0 & (\frac{\partial}{\partial\mathbf{h}}\mathcal{L}(\mathbf{h},\mathbf{y},\xi) = 0) \\ \mathbf{1}^\top\mathbf{h} = 1 & \text{(primal feasibility)} \\ 0 \leq \mathbf{y} \perp \mathbf{h} \geq 0 & \text{(complementary slackness).} \end{array}$$

It is not hard to see that the following is a solution satisfying the KKT condition

$$\begin{array}{rl} \mathbf{y} & = (\mathbf{x}+\xi\mathbf{1})_+ - (\mathbf{x}+\xi\mathbf{1}) \\ \mathbf{h} & = (\mathbf{x}+\xi\mathbf{1})_+ \\ \xi & = \text{the solution of } \sum_{i=1}^n (x_i+\xi)_+ = 1. \end{array}$$

Here we define $(x)_+ = \max(x, 0)$ is the hinge function.

Proof of Theorem 3. To count the total number of different gamma correction transforms, we notice that if i is mapped to j by the gamma correction transform, we have

$$j - \frac{1}{2} \leq n\left(\frac{i}{n}\right)^\gamma < j + \frac{1}{2}.$$

This turns into $\underline{\gamma}_{ij} \leq \gamma < \bar{\gamma}_{ij}$ where

$$\underline{\gamma}_{ij} = \frac{\log n - \log(j+\frac{1}{2})}{\log n - \log i}, \bar{\gamma}_{ij} = \frac{\log n - \log(j-\frac{1}{2})}{\log n - \log i}.$$

We have (i) all $\underline{\gamma}_{ij}$ and $\bar{\gamma}_{ij}$ are distinct numbers and (ii) $\underline{\gamma}_{ij} = \bar{\gamma}_{i,j+1}$. As such, each different value of $\underline{\gamma}_{ij}$ signifies a change of gamma correction curves, while different values within the range $\underline{\gamma}_{ij} \leq \gamma < \bar{\gamma}_{ij}$ corresponds to the same curve. Therefore, the total number of gamma correction is bounded by $(n-1)^2$.

Optimization of (13). Using Theorem 2, we introduce two auxiliary variables $\hat{\mathbf{u}} \geq 0$ and $\hat{\mathbf{v}} \geq 0$ to replace the ℓ_1 norms in (13), and reformulate the problem as

ALGORITHM 4: Optimization of (13)

initialize $\hat{\mathbf{h}}_0$, $\hat{\mathbf{u}}_0$ and $\hat{\mathbf{v}}_0$, $t \leftarrow 0$
while not converge **do**
  $\hat{\mathbf{h}}_{t,0} \leftarrow \hat{\mathbf{h}}_t$, $\tau \leftarrow 0$
  while not converge **do**
    $\hat{M} \leftarrow (RF)^\top \mathcal{D}(\hat{\mathbf{u}}_t)^{-1} RF + \xi F^\top \mathcal{D}(\hat{\mathbf{v}}_t)^{-1} F$
    $\hat{\mathbf{b}} \leftarrow (RF)^\top \mathcal{D}(\hat{\mathbf{u}}_t)^{-1} F\tilde{\mathbf{h}} + \xi F^\top \mathcal{D}(\hat{\mathbf{v}}_t)^{-1} F T_\phi \mathbf{h}$
    $\triangle \hat{\mathbf{h}}_\tau \leftarrow \hat{M}\hat{\mathbf{h}}_{t,\tau} - \hat{\mathbf{b}}$
    $\hat{\mathbf{h}}_{t,\tau+1} \leftarrow \mathcal{P}_{\Delta^{n+1}}\left(\hat{\mathbf{h}}_{t,\tau} - \frac{\eta_0}{\tau+1}\triangle\hat{\mathbf{h}}_\tau\right)$, $\tau \leftarrow \tau + 1$
  end while
  $\hat{\mathbf{h}}_{t+1} \leftarrow \hat{\mathbf{h}}_{t,\tau}$, $\hat{\mathbf{u}}_{t+1} \leftarrow \left|F\left(\tilde{\mathbf{h}} - R\hat{\mathbf{h}}_{t+1}\right)\right|$;
  $\hat{\mathbf{v}}_{t+1} = \left|F\left(\hat{\mathbf{h}}_{t+1} - T_\phi \mathbf{h}\right)\right|$; $t \leftarrow t + 1$;
end while

$$\min_{\hat{\mathbf{h}},\hat{\mathbf{u}},\hat{\mathbf{v}}} \hat{L}(\hat{\mathbf{h}},\hat{\mathbf{u}},\hat{\mathbf{v}}) \text{ s.t. } \mathbf{1}^T\hat{\mathbf{h}} = 1, \hat{\mathbf{h}} \geq 0, \hat{\mathbf{u}} \geq 0, \hat{\mathbf{v}} \geq 0, \tag{24}$$

where

$$\hat{L}(\hat{\mathbf{h}},\hat{\mathbf{u}},\hat{\mathbf{v}}) = \frac{1}{2}\mathbf{1}^\top\hat{\mathbf{u}} + \frac{\xi}{2}\mathbf{1}^\top\hat{\mathbf{v}} + \frac{1}{2}\left(F\tilde{\mathbf{h}} - FR\hat{\mathbf{h}}\right)^\top \mathcal{D}(\hat{\mathbf{u}})^{-1}\left(F\tilde{\mathbf{h}} - FR\hat{\mathbf{h}}\right) + \frac{\xi}{2}\left(F\hat{\mathbf{h}} - FT_\phi\mathbf{h}\right)^\top \mathcal{D}(\hat{\mathbf{v}})^{-1}\left(F\hat{\mathbf{h}} - FT_\phi\mathbf{h}\right).$$

Eq.(25) is a convex optimization problem jointly for $(\hat{\mathbf{h}},\hat{\mathbf{u}},\hat{\mathbf{v}})$, and we solve it also with a block coordinate descent scheme. Specifically, initializing $\hat{\mathbf{h}}_0$, $\hat{\mathbf{u}}_0$ and $\hat{\mathbf{v}}_0$, we find the optimal solution to it by iterating the following steps until convergence

- $\hat{\mathbf{h}}_{t+1} \leftarrow \text{argmax}_{\hat{\mathbf{h}}}\, \hat{L}(\hat{\mathbf{h}},\hat{\mathbf{u}}_t,\hat{\mathbf{v}}_t)$;
- $\hat{\mathbf{u}}_{t+1} \leftarrow \text{argmax}_{\hat{\mathbf{u}}}\, \hat{L}(\hat{\mathbf{h}}_{t+1},\hat{\mathbf{u}},\hat{\mathbf{v}}_t)$;
- $\hat{\mathbf{v}}_{t+1} \leftarrow \text{argmax}_{\hat{\mathbf{v}}}\, \hat{L}(\hat{\mathbf{h}}_{t+1},\hat{\mathbf{u}}_{t+1},\hat{\mathbf{v}})$.

Optimizing $\hat{\mathbf{h}}$: fixing $\hat{\mathbf{u}}_t$ and $\hat{\mathbf{v}}_t$ and dropping irrelevant terms, minimizing $\hat{\mathbf{h}}$ reduces to the following constrained linear least squares problem

$$\min_{\hat{\mathbf{h}}} \frac{1}{2}\hat{\mathbf{h}}^\top\hat{M}\hat{\mathbf{h}} - \hat{\mathbf{b}}^\top\hat{\mathbf{h}} \text{ s.t. } \mathbf{1}^T\hat{\mathbf{h}} = 1, \hat{\mathbf{h}} \geq 0, \tag{25}$$

where

$$\hat{M} = (RF)^\top \mathcal{D}(\hat{\mathbf{u}}_t)^{-1} RF + \xi F^\top \mathcal{D}(\hat{\mathbf{v}}_t)^{-1} F,\ \hat{\mathbf{b}} = (RF)^\top \mathcal{D}(\hat{\mathbf{u}}_t)^{-1} F\tilde{\mathbf{h}} + \xi F^\top \mathcal{D}(\hat{\mathbf{v}}_t)^{-1} FT_\phi\mathbf{h}.$$

Eq.(25) is solved with projected gradient descent: starting with $\mathbf{h}_{t,0} = \mathbf{h}_t$, our algorithm iterates between two steps:

$$\triangle\hat{\mathbf{h}}_\tau \leftarrow \hat{M}\hat{\mathbf{h}}_{t,\tau} - \hat{\mathbf{b}},\ \hat{\mathbf{h}}_{t,\tau+1} \leftarrow \mathcal{P}_{\Delta^{n+1}}\left(\hat{\mathbf{h}}_{t,\tau} - \frac{\eta_0}{\tau+1}\triangle\hat{\mathbf{h}}_\tau\right). \tag{26}$$

We take $\hat{\mathbf{h}}_{t+1} = \hat{\mathbf{h}}_{t,\tau}$ at the convergence.

Optimizing $\hat{\mathbf{u}}$: when fixing $\hat{\mathbf{h}}_{t+1}$ and dropping irrelevant terms, minimizing $\hat{\mathbf{u}}$ becomes

$$\min_{\hat{\mathbf{u}}} \mathbf{c}^\top\mathcal{D}(\hat{\mathbf{u}})^{-1}\mathbf{c} + \mathbf{1}^\top\hat{\mathbf{u}} \text{ s.t. } \hat{\mathbf{u}} \geq 0,$$

where $\mathbf{c} = F\left(\tilde{\mathbf{h}} - R\hat{\mathbf{h}}_{t+1}\right)$. Using Theorem 2, we obtain the optimal solution $\hat{\mathbf{u}}_{t+1} \leftarrow \left|F\left(\tilde{\mathbf{h}} - R\hat{\mathbf{h}}_{t+1}\right)\right|$.

Optimizing $\hat{\mathbf{v}}$: similarly, when fixing $\hat{\mathbf{h}}_{t+1}$ and dropping irrelevant terms, minimizing $\hat{\mathbf{v}}$ becomes

$$\min_{\hat{\mathbf{v}}} \left(\hat{\mathbf{h}}_{t+1} - T_\phi\mathbf{h}\right)^\top F^\top\mathcal{D}(\hat{\mathbf{v}})^{-1}F\left(\hat{\mathbf{h}}_{t+1} - T_\phi\mathbf{h}\right) + \mathbf{1}^\top\hat{\mathbf{v}},\ \text{s.t. } \hat{\mathbf{v}} \geq 0.$$

Using Theorem 2 again, we have $\hat{\mathbf{v}}_{t+1} = \left|F\left(\hat{\mathbf{h}}_{t+1} - T_\phi\mathbf{h}\right)\right|$.

The overall algorithm is given in the pseudo-code given in Algorithm 4.

REFERENCES

[1] M. Barni, M. Fontani, and B. Tondi. 2012. A universal technique to hide traces of histogram-based image manipulations. In *ACM Workshop on Multimedia and Security*.

[2] Dimitri P. Bertsekas. 1996. *Constrained optimization and Lagrange multiplier methods,*. Athena Scientific.

[3] S. Boyd and L. Vandenberghe. 2005. *Convex Optimization*. Cambridge University Press.

[4] Yuri Boykov, Olga Veksler, and Ramin Zabih. 2001. Fast Approximate Energy Minimization via Graph Cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23, 11 (2001), 1222–1239.

[5] Gang Cao, Yao Zhao, and Rongrong Ni. 2010. Forensic estimation of gamma correction in digital images. In *IEEE International Conference on Image Processing (ICIP)*. https://doi.org/10.1109/ICIP.2010.5652701

[6] Gang Cao, Yao Zhao, Rongrong Ni, and Xuelong Li. 2014. Contrast Enhancement-Based Forensics in Digital Images. *IEEE Transactions on Information Forensics and Security* 9, 3 (2014), 515–525.

[7] G. Cao, Y. Zhao, R. Ni, and H. Tian. 2010. Anti-forensics of contrast enhancement in digital images. In *ACM Workshop on Multimedia and Security*.

[8] CVX Research Inc. 2012. CVX: Matlab Software for Disciplined Convex Programming, version 2.0 beta. (2012).

[9] Duc-Tien Dang-Nguyen, Cecilia Pasquini, Valentina Conotter, and Giulia Boato. 2015. RAISE - A Raw Images Dataset for Digital Image Forensics. In *ACM Multimedia Systems*. Portland, OR.

[10] H. Farid. 2001. Blind Inverse Gamma Correction. *IEEE Transactions On Image Processing* 10, 10 (2001), 1102–1125.

[11] Hany Farid (Ed.). 2016. *Photo Forensics*. MIT Press.

[12] P. Ferrara, T. Bianchiy, A. De Rosaz, and A. Piva. 2013. Reverse engineering of double compressed images in the presence of contrast enhancement. In *IEEE Workshop Multimedia Signal Processing*. Pula, Croatia.

[13] M. Gonzalez and F. Woods. 2002. *Digital image processing* (2nd ed.). Printice Hall.

[14] Ralph P Grimaldi. 1998. *Discrete and Combinatorial Mathematics: An Applied Introduction* (4th ed.).

[15] E. Levina and P. Bickel. 2001. The Earth Mover's distance is the Mallows distance: some insights from statistics. In *IEEE International Conference on Computer Vision*, Vol. 2. 251–256 vol.2. https://doi.org/10.1109/ICCV.2001.937632

[16] Xufeng Lin, Xingjie Wei, and Chang-Tsun Liang. 2014. Two improved forensic methods of detecting contrast enhancement in digital images. In *SPIE 9028, Media Watermarking, Security, and Forensics*.

[17] Siwei Lyu, Xunyu Pan, and Xing Zhang. 2014. Exposing Region Splicing Forgeries with Blind Local Noise Estimation. *International Journal of Computer Vision* 110, 2 (November 2014), 202–221.

[18] Jean Morales, Charles A. Micchelli, and Massimiliano Pontil. 2010. A Family of Penalty Functions for Structured Sparsity. In *Advances in Neural Information Processing Systems*.

[19] M.C. Stamm and K.J.R. Liu. 2010. Forensic detection of image manipulation using statistical intrinsic fingerprints. *IEEE Transactions on Information Forensics and Security* 5, 3 (2010), 492–506.

[20] M.C. Stamm and K.J.R. Liu. 2010. Forensic estimation and reconstruction of a contrast enhancement mapping. In *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*. 1698 –1701. https://doi.org/10.1109/ICASSP.2010.5495488

[21] C Villani. 2008. *Optimal Transport: Old and New*. Springer, Berlin.

[22] Xing Zhang and Siwei Lyu. 2014. Blind Estimation Of Pixel Brightness Transform. In *IEEE Conference on Image Processing (ICIP)*. Paris, France.

[23] X. Zhang and S. Lyu. 2014. Estimating Covariance Matrix Of Correlated Image Noise. In *IEEE Conference on Image Processing (ICIP)*. Paris, France.

[24] Daniel Zoran and Yair Weiss. 2009. Scale Invariance and Noise in Nature Image. In *IEEE International Conference on Computer Vision*. Kyoto, Japan.