BLIND ESTIMATION OF PIXEL BRIGHTNESS TRANSFORM

Xing Zhang and Siwei Lyu

Computer Science Department
University at Albany, State University of New York

ABSTRACT

Pixel brightness transforms (PBT), examples of which include gamma correction, sigmoid stretching and histogram equalization, are common operations on digital images, and it is practically useful to estimate such transforms directly from an image. In this work, we describe an effective and efficient method to estimate PBT from images, which takes advantage of the nature of PBT as a mapping between integral pixel values and the distinct characteristics it introduces to the pixel value (PV) histograms of the transformed image. Our method recovers the original PV histogram and the PBT 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.

Index Terms— blind estimation, pixel brightness transform, pixel histogram

1. INTRODUCTION

The integral pixel values in a digital image do not directly reflect the amount of light projected on the camera sensor at the time of capture. Such is mainly due to the two mappings in the imaging process. First, the continuous scene radiance \( x \) is mapped to raw integral pixel value \( j = \psi(x) \) with the camera response function (CRF). Subsequently the raw pixel values may further undergo the Pixel brightness transform (PBT) to become the final integral pixel value \( i = \phi(j) \). Examples of PBT include gamma correction, sigmoid stretching or histogram equalization [7].

From the practical point of view, blind estimation of PBT from a single image is useful for two main reasons. First, it can help to correct nonlinearity applied to integral pixel values, and thus facilitates radiometric calibration and other computer vision applications, such as image registration, matching and retrieval, that predicate on raw pixel values. Second, in forensic analysis of digital images [2], recovering PBT can be used to reconstruct the processing history of an image in establishing its authenticity, or to differentiate regions from different source images in detecting tampering.

There have been only a few previous works addressing this problem. As the most common PBT, several works have focused on the blind estimation of gamma correction. The method in [8] uses higher order statistical dependencies introduced by gamma. The method of [2] uses the the features developed in [2] 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 PV histogram features. Using a set of features characterizing the shapes of PV histogram after a PBT, the work in [2] develops method to determine if an image has undergone gamma correction or histogram equalization, but this method cannot recover the actual transform. The work [2] describes an iterative algorithm to jointly estimate a nonparametric PBT and the PV histogram of the original image, based on a probabilistic model of PV histogram and an exhaustive matching procedure to determine which histogram entries are most likely to correspond to artifacts caused by the PBT. Last, there are recent works (e.g., [2]) to recover the whole image processing pipeline including PBT blindly from images.

In this work, we describe an effective and efficient method to estimate PBT from a single image. This method takes advantage of the special nature of PBT as a mapping between integral pixel values that introduces distinct characteristics to the pixel value (PV) histograms of the transformed image. It recovers the original PV histogram and the PBT simultaneously with an efficient iterative algorithm. We show experimental results demonstrating the efficacy and efficiency of our method.

2. PBT AND ITS EFFECT ON PV HISTOGRAM

For simplicity, we focus our discussion on gray-scale images whose pixel values are encoded with \( b \)-bits that correspond to \( N = 2^b \) different values. Most PBTs can be modeled as a continuous mapping followed by a rounding operation that maps real value back to integers\(^1\). Specifically, each input pixel value, \( j \in \{0, \cdots, N-1\} \), is mapped to its corresponding output pixel value, \( i \in \{0, \cdots, N-1\} \), as \( i = \phi(j) := \text{round}[m(j)], \) where \( m(\cdot) : [0,N-1] \mapsto [0,N-1] \) is a continuous transform. Furthermore, most forms of \( m(\cdot) \) of practical interest is monotonic, which we will assume subse-

\(^1\)Other operations recovering integer values from continuous values such as floor or ceiling can also be used.
that is defined as
form, as:
undergone PBTs.  
original image is considerably smaller than those of images
ence of neighboring components for the PV histogram of the
range of other photographic images.
sequently. Examples of \( m(\cdot) \) include
- gamma correction,
  \[
m_\gamma(j) = N \left( \frac{j}{N} \right)^\gamma,
  \]
- sigmoid stretching
  \[
m_{\alpha,\beta}(j) = N \left( 1 + \exp \left( \frac{\beta - \frac{j}{N}}{\alpha} \right) \right)^{-1},
  \]
- histogram equalization,
  \[
m(j) = \rho^{-1} \left( \frac{j}{N} \right),
  \]
where \( \rho^{-1}(\cdot) \) is the interpolated inverse cumulative
density function of the pixels of the input image,
- free-form intensity transform, such as the curves tool
  in Adobe Photoshop.

The first two are examples of parametric PBT and the latter
two are nonparametric PBTs. It should also be pointed out
that even though \( m(\cdot) \) is invertible, the overall PBT in general
may not be invertible due to the rounding step.

Because of its Pixel nature, we can understand the effect
of PBT on an image by examining the pixel value (PV) his-
togram. Specifically, we have
\[
\mathcal{P}_{\text{output}}(i) = \sum_{j=0}^{N-1} \delta \left( i = \text{round}[m(j)] \right) \cdot \mathcal{P}_{\text{input}}(j).
\]
For convenience, we normalize the PV histogram to have
a unit total sum, so it can be treated as a probability dis-
tribution over pixel values. We represent the normalized
PV histograms of the input and output image as vectors
on the \( N \)-dimensional probability simplex, such that \( x_j = \mathcal{P}_{\text{input}}(j - 1) \) and \( y_i = \mathcal{P}_{\text{output}}(i - 1) \), and introduce matrix
\( H \) that is defined as \( H_{ij} = \delta \left( i - \text{round}[m(j - 1)] + 1 \right) \),
for \( i,j \in \{1, \cdots, N\} \), then Eq.\((?)\) can be rewritten as
\( y_i = \sum_{j=1}^{N} H_{ij} x_j \), or in the more compact matrix and vector
form, as:
\[
y = H x. \tag{5}
\]
For PBT that is not identity, there will be multiple input
pixel values map to a single output pixel value, and output
pixel values to which no input pixel value maps based on the
pigeonhole principle [?]. This redistribution of pixel values
creates peaks and gaps in the shape of the output PV his-
togram, which is illustrated in two cases (gamma correction
and histogram equalization) in the second and third rows of
Fig.\(\)??. In contrast, the PV histogram of typical photographic
image not undergoing PBT is more smooth (first row Fig.\(\)??).

The difference in smoothness of these PV histograms can be
better revealed with the first order difference of the values of
neighboring bins, as shown in the last column of Fig.\(\)??. Note
that when measured with the \( \ell_1 \) norm, the first order differ-
ence of neighboring components for the PV histogram of the
original image is considerably smaller than those of images
undergone PBTs\(^2\).

\(^2\) Similar regularities of PV histograms have also been observed on a wide
range of other photographic images.

\[\text{Fig. 1: Effects of Pixel brightness transform (PBT) on a photo-
graphic image and its pixel value (PV) histogram. The three rows in
the top panel correspond to the three rows in the bottom.}\]

3. ESTIMATING PBT FROM IMAGE

The central problem in PBT estimation is to recover the trans-
form and the vectorized PV histogram \( x \) of the input image
using only the vectorized PV histogram of an image under-
gone PBT, \( y \). More specifically, this is to solve for \( H \) and \( x \)
simultaneously in Eq.\((?)\) with known \( y \).

3.1. Recovering Original PV Histogram

The problem of recovering \( x \) from \( y \) and \( H \) is oftentimes ill-
posed – unless matrix \( H \) is identity, it contains rows of all
zeros corresponding to output pixel values to which no input
pixel value maps.

To obtain a stable solution, we need to enforce more con-
straints on the values of \( x \). As results in Fig.\(\)?? suggest, we
formulate the problem of estimating \( x \) as an \( \ell_1 \) constrained
least squares problem – we seek the optimal \( x \) that leads to
“similar” PV histogram to \( y \) measured by the \( \ell_2 \) difference,
and has small \( \ell_1 \) norm for the first order differences of neigh-
boring components. Formally, this is expressed with the fol-
lowing constrained optimization problem:
\[
\min_{x, \lambda: \ell_1 x = 1, x \geq 0} \frac{1}{2} \|y - H x\|^2 + \lambda \|D x\|_1 \tag{6}
\]
where \( \lambda \geq 0 \) is a system parameter that controls the trade-off between the \( \ell_2 \) and \( \ell_1 \) losses. Matrix \( D \) corresponds to the first order discrete difference operator. We will refer to the optimal value of the objective function of (18) as \( L^*(H, \lambda) \). The optimization problem in (18) is convex, and it can be solved with general-purpose convex programming package such as \textsc{cvx} [?, ?]. Shown in Fig.2 is an example of the recovery of the PV histogram of an 8-bit grayscale image undergone a gamma correction with \( \gamma = 0.4 \) (the corresponding matrix \( H \) has a rank of 176 out of a full rank of 256). As the result shows, the recovered PV histogram is very close to the original PV histogram, and the presence of small flat regions are the results of the smoothness penalty.

3.2. Recovering Pixel Brightness Transform

We now turn to the problem of estimating the original PV histogram \( x \) and transformation matrix \( H \) simultaneously. We discuss two cases: (i) parametric PBT specified with a small number of parameters and (ii) nonparametric PBT as a mapping for every possible input in \( \{0, \cdots, N-1\} \).

**Parametric PBT.** We have consistently observed that the true transform parameter usually corresponds to a global minimum of the objective function of (18) over a range of values. For instance, Fig.2 shows the values of \( L^*(H(\gamma), \lambda) \) corresponding to gamma correction PBT, Eq.(18), with \( \gamma \in [0.1, 1.8] \) (\( \lambda = 0.1 \) is fixed in all cases). The two curves shown correspond to \( \gamma = 0.4 \) and 1.4, respectively, where the true parameter values lead to global minimums of \( L^*(H(\gamma), \lambda) \) as shown by the red circles.

![Fig. 2: Recovery of the PV histogram of an 8-bit grayscale image that has been gamma corrected with \( \gamma = 0.4 \) (the corresponding matrix \( H \) has a rank of 176 out of a full rank of 256). (left) original PV histogram. (middle) observed PV histogram after gamma correction. (right) recovered PV histogram.](image)

Based on this observation, our method of finding the optimal PBT parameter is based on solving

\[
\gamma^* = \arg\min_{\gamma} L^*(H(\gamma), \lambda),
\]

where we use \( H(\gamma) \) to make explicit the dependence of matrix \( H \) on parameter \( \gamma \). This objective function is not differentiable with regards to \( \gamma \) due to the rounding step. Yet, when the PBT can be specified by one or two parameters, we found a simpler approach, which first performs a grid search in a given range of parameter values followed by a binary search between the reduced ranges where overall values of \( L^*(H(\gamma), \lambda) \) is minimal, is usually effective and leads to satisfactory solutions to (18).

**Nonparametric PBT.** When the PBT cannot be specified with a parametric form, we estimate the PBT directly and recover the original PV histogram simultaneously. This is achieved with an iterative method: starting with initial values for \( H \) and \( x \), we iterate between (i) solving for \( x \) with fixed \( H \) and (ii) solving \( H \) with fixed \( x \). The former is solved with the steps described in Section 2, and we turn to the solution of the latter here. Specifically, with the vectorized input and output PV histograms \( x \) and \( y \), it can be shown that a unique deterministic monotonic transform exists that satisfies (i) it converts a variable with distribution given by \( x \) to another that has distribution given by \( y \), and (ii) it minimizes the overall work that is measured by the total displacement of bins.

Formally, denote \( \chi_x(\cdot) : \{0, N-1\} \rightarrow [0, 1] \) and \( \chi_y(\cdot) : \{0, N-1\} \rightarrow [0, 1] \) as the cumulative density functions induced from \( x \) and \( y \), respectively. i.e., \( \chi_x(i) = \sum_{k=1}^{i} x_k \) and \( \chi_y(i) = \sum_{k=1}^{i} y_k \), and \( \chi^{-1}_y(\cdot) : [0, 1] \rightarrow \{0, N-1\} \) as the pseudo inverse cumulative distribution induced from \( y \), as \( \chi^{-1}_y(p) = i \) if \( \chi_y(i-1) < p \leq \chi_y(i) \). Then the mapping satisfying the two aforementioned conditions is given by [?]

\[
f(i) = \chi^{-1}_y(\chi_x(i)),
\]

and the corresponding transform matrix is obtained as \( H_{ij} = \delta(j = f(i)) \).

In summary, our algorithm iterates between optimizing \( x \) with (18) and computing mapping from histogram matching with Eq.(18). While we have not obtain a theoretical proof of convergence, in practice we observe that the algorithm usually converges within less than 10 iterations. Fig.2 demonstrate the convergence of one estimated PBT, with the original PBT obtained from cubic spline interpolation of key points that are chosen manually. As it shows, after 5 iterations of the algo-

---

*Similar observations have also been made on other types of parametric PBTs, such as sigmoid stretching and cubic spline curves.

*This is a direct corollary of the optimal transportation plan on the real line in the Monge-Kantorovitch transportation theory framework [?].
We perform several experiments to evaluate the described method. The data used in our experiments are 100 grayscale images from the van Hateren database. We estimate the parameters of several parametric PBTs, including gamma correction, sigmoid stretching, and free-form stretching, as described in Section 2. The latter resembles tone mapping tools in photo-editing software (e.g., the Curve tool in Photoshop). We applied histogram equalization and free-form PBTs to each of the test images, and then used the algorithm described in Section 2 to recover the integral mapping in the PBT directly. We measure the performance by the root mean squared error (RMSE) between the actual PBTs and the corresponding estimated PBTs, treating both as functions over $[0, N - 1] \rightarrow [0, N - 1]$. We compare with the only known previous work for the same task is [2], which is based on an iterative and exhaustive search of PV histograms that can lead to the observed PV histogram of an image with PBT. As the results in Table ?? show, both algorithms achieve good estimation performances when there is no perturbation. However, our method on average is $5 \times 10$ times faster in running time, because the method [2] relies on exhaustive search.

### 5. CONCLUSION

In this work, we have described an effective method that can blindly recover Pixel brightness transforms as mappings between integral pixel intensities. Our method takes advantage of the observed smoothness of PV histograms of the untransformed digital images. We demonstrate the effectiveness of our method on a set of images and showing recovery results close to perfect. There are several extensions we would like to further study. First, the method described here can be extended to the cases of RGB color images. Furthermore, by examining local areas in the image, we will also investigate using this method for the detection of forged image regions that have undergone different PBTs.

**Acknowledgment:** This work is partly supported by the National Science Foundation under Grant Nos. IIS-0953373, IIS-1208463 and CCF-1319800. We also thank the anonymous reviewers for their helpful comments.
6. REFERENCES


