

Computing a Non-trivial Lower Bound on the Joint Entropy of Two Images



**Approved for public release.
Distribution is unlimited.**

Kalyan S. Perumalla

March 10, 2017

DOCUMENT AVAILABILITY

Reports produced after January 1, 1996, are generally available free via US Department of Energy (DOE) SciTech Connect.

Website: <http://www.osti.gov/scitech/>

Reports produced before January 1, 1996, may be purchased by members of the public from the following source:

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
Telephone: 703-605-6000 (1-800-553-6847)
TDD: 703-487-4639
Fax: 703-605-6900
E-mail: info@ntis.gov
Website: <http://classic.ntis.gov/>

Reports are available to DOE employees, DOE contractors, Energy Technology Data Exchange representatives, and International Nuclear Information System representatives from the following source:

Office of Scientific and Technical Information
PO Box 62
Oak Ridge, TN 37831
Telephone: 865-576-8401
Fax: 865-576-5728
E-mail: report@osti.gov
Website: <http://www.osti.gov/contact.html>

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Computational Sciences and Engineering Division

Computing a Non-trivial Lower Bound on the Joint Entropy of Two Images

Kalyan S. Perumalla

Date Published: March 2017

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, TN 37831-6283
managed by
UT-Battelle, LLC
for the
US DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

CONTENTS

ABSTRACT	1
1. NOTATION AND DEFINITIONS	1
2. ALGORITHMS TO COMPUTE JOINT ENTROPY LOWERBOUND	4
2.1 RUNTIME COMPLEXITY	5
2.2 MAIN ALGORITHM	5
2.3 ALTERNATIVE LOOP OF THE MAIN ALGORITHM	7
2.4 HEURISTIC VARIANT OF THE MAIN ALGORITHM	8
3. CONCLUDING REMARKS	8
3.1 SOFTWARE IMPLEMENTATION	8
3.2 POTENTIAL APPLICATION	9
4. REFERENCES	9

ABSTRACT

In this report, a non-trivial lower bound on the joint entropy of two non-identical images is developed, which is greater than the individual entropies of the images. The lower bound is the least joint entropy possible among all pairs of images that have the same histograms as those of the given images. New algorithms are presented to compute the joint entropy lower bound with a computation time proportional to $S \log S$ where S is the number of histogram bins of the images. This is faster than the traditional methods of computing the exact joint entropy with a computation time that is quadratic in S .

1. NOTATION AND DEFINITIONS

The notation, definitions, and some key properties are introduced in this section, followed by Section 2. containing three algorithms to compute a non-trivial lower bound on joint entropy of two images. Potential application of the algorithms is indicated in Section 3..

Let $I(H \times W)$ denote an image I of height H and width W . Let $I[i][j]$ denote the color shade value of the image pixel at (i, j) , $0 \leq i < H$ and $0 \leq j < W$. Let S be the total number of unique shade values ($0 \leq I[i][j] < S$).

- **1-D Histogram** The one-dimensional (1-D) histogram b is an array of counters $b[0..S - 1]$ such that $0 \leq b[s] = c_s$, $0 \leq s < S$, where $C_b = \sum_{s=0}^{S-1} b[s]$.

- **Reach of a 1-D Histogram** The reach R_b of a 1-D histogram b is the number of non-zero bins in b :

$$R_b = |\{s | b[s] > 0, 0 \leq s < S\}| . \quad (1)$$

- **Entropy of a 1-D Histogram** The entropy $EH(b)$ of a 1-D histogram b is given as

$$\begin{aligned} EH(b) &= - \sum_{s=0}^{S-1} p_s \log p_s \quad \text{for all } p_s > 0, \text{ where} \\ p_s &= b[s]/C_b \quad 0 \leq s < S . \end{aligned} \quad (2)$$

- **1-D Histogram of an Image** The 1-D histogram b_I of an image $I(H \times W)$ is defined by

$$\begin{aligned} b_I[s] &= |\{(i, j) | I[i][j] = s\}| \quad \text{for all } 0 \leq s < S, \text{ where} \\ \sum_{s=0}^{S-1} b_I[s] &= HW = C_{b_I} . \end{aligned} \quad (3)$$

- **Entropy of an Image** The entropy $E(I)$ of an image $I(H \times W)$ is defined as the entropy of its 1-D histogram b_I :

$$E(I) = EH(b_I) . \quad (4)$$

- **2-D Histogram** The two-dimensional (2-D) histogram B is an array of counters $B[0..S - 1][0..S - 1]$ such that $0 \leq B[s_1][s_2] = c_{s_1, s_2}$, where $C_B = \sum_{s_1=0}^{S-1} \sum_{s_2=0}^{S-1} B[s_1][s_2]$.

- **Reach of a 2-D Histogram** The reach R_B of a 2-D histogram B is the number of non-zero bins in B :

$$R_B = |\{x | x = B[s_1][s_2] \text{ and } x > 0, 0 \leq s_1, s_2 < S\}|. \quad (5)$$

- **Entropy of a 2-D Histogram** The entropy $EH(B)$ of a 2-D histogram B is given as

$$\begin{aligned} EH(B) &= - \sum_{s_1=0}^{S-1} \sum_{s_2=0}^{S-1} p_{s_1 s_2} \log p_{s_1 s_2}, \text{ for } p_{s_1 s_2} > 0, \text{ where} \\ p_{s_1 s_2} &= B[s_1][s_2]/C_B, 0 \leq s_1, s_2 < S. \end{aligned} \quad (6)$$

- **2-D Histogram of Two Images** The 2-D histogram $B_{I_1 I_2}$ of two images $I_1(H \times W)$ and $I_2(H \times W)$ is defined by

$$\begin{aligned} B_{I_1 I_2}[s_1][s_2] &= |\{(i, j) | I_1[i][j] = s_1 \text{ and } I_2[i][j] = s_2\}|, \text{ where} \\ \sum_{s_1=0}^{S-1} \sum_{s_2=0}^{S-1} B_{I_1 I_2}[s_1][s_2] &= HW = C_{B_{I_1 I_2}}. \end{aligned} \quad (7)$$

- **Minimal 2-D Histogram of Two Images** The minimal 2-D histogram $B^*(I_1, I_2)$ of two images I_1 and I_2 is defined by

$$\begin{aligned} B^*(I_1, I_2) &= \text{histogram } B \text{ with minimal reach } R_B \text{ among all possible } B_{\hat{I}_1 \hat{I}_2}, \\ \text{such that } b_{\hat{I}_1} &= b_{I_1} \\ \text{and } b_{\hat{I}_2} &= b_{I_2} \end{aligned} \quad (8)$$

In other words, $B^*(I_1, I_2)$ is a 2-D histogram that has the smallest reach R_{B^*} among all possible 2-D histograms $B_{\hat{I}_1 \hat{I}_2}$ for any images \hat{I}_1 and \hat{I}_2 that have the same histograms as I_1 and I_2 respectively, that is, $b_{\hat{I}_1} = b_{I_1}$ and $b_{\hat{I}_2} = b_{I_2}$.

- **Joint Entropy of Two Images** The joint entropy $JE(I_1, I_2)$ between two equal-sized images $I_1(H \times W)$ and $I_2(H \times W)$ is defined as the entropy of their joint 2-D histogram $B_{I_1 I_2}$:

$$JE(I_1, I_2) = EH(B_{I_1 I_2}). \quad (9)$$

- **Lower Bound on the Joint Entropy of Two Images** The non-trivial joint entropy lower bound $JE_{LB}(I_1, I_2)$ between two equal-sized, non-identical images $I_1(H \times W)$ and $I_2(H \times W)$ is defined as

$$\begin{aligned} JE_{LB}(I_1, I_2) &\leq JE(\hat{I}_1, \hat{I}_2), \\ \text{among all possible image pairs } \langle \hat{I}_1, \hat{I}_2 \rangle &\text{ with the same histograms as of } I_1 \text{ and } I_2, \\ \text{that is, } b_{\hat{I}_1} &= b_{I_1} \text{ and } b_{\hat{I}_2} = b_{I_2}. \end{aligned} \quad (10)$$

- **Normalized Mutual Information of Two Images** Given two images I_1 and I_2 of the *same size*, the normalized mutual information, $NMI(I_1, I_2)$, between the two images is defined as

$$NMI(I_1, I_2) = \frac{E(I_1) + E(I_2)}{JE(I_1, I_2)}. \quad (11)$$

- **An Upper Bound on Normalized Mutual Information of Two Images** Given two images I_1 and I_2 of the *same size*, a (non-trivial) upper bound, $NMI_{UB}(I_1, I_2)$, on the normalized mutual information, $NMI(I_1, I_2) < 2$, between the two images is defined as

$$NMI_{UB}(I_1, I_2) > NMI(I_1, I_2). \quad (12)$$

- **Maximal Normalized Mutual Information of Two Images** Let $I_g(h \times w)$ be an image no bigger than another image $I_G(H \times W)$, $h \leq H$ and $w \leq W$. Let $I_{G_{\hat{i}\hat{j}}}$ be the $h \times w$ sub-image of I_G starting at (\hat{i}, \hat{j}) , $0 \leq \hat{i} \leq H - h$ and $0 \leq \hat{j} \leq W - w$. The maximal mutual entropy of the two images is defined as

$$\begin{aligned} MNMI &= \max \left(NMI(I_g, I_{G_{\hat{i}\hat{j}}}), 0 \leq \hat{i} \leq H - h, 0 \leq \hat{j} \leq W - w \right) \\ &\leq \max \left(NMI_{UB}(I_g, I_{G_{\hat{i}\hat{j}}}), 0 \leq \hat{i} \leq H - h, 0 \leq \hat{j} \leq W - w \right). \end{aligned} \quad (13)$$

Lemma 1. *Increasing the number of non-zero bins in any given histogram increases its entropy.*

Proof. Let b be the original histogram and b' be the modified histogram obtained by a sub-division of b . Consider the smallest possible sub-division, namely, smearing the count of one shade across two shades, leaving all the other counts unchanged. In other words, one of the non-zero bins is being sub-divided into two non-zero bins. Pick any element $b[s]$ to be divided from one shade s into two shades s_1 and s_2 , where $b[s] = b[s_1] + b[s_2]$. Thus, b' differs from b only in the aforementioned modification of $b[s]$, $b[s_1]$ and $b[s_2]$. Note that either s_1 or s_2 could be the same as s , but not both.

Let $C = \sum_{s=0}^{S-1} b[s]$ be the total count of elements in the histogram. Let $p = \frac{b[s]}{C}$, $p_1 = \frac{b[s_1]}{C}$, and $p_2 = \frac{b[s_2]}{C}$. Note that $p = p_1 + p_2$, $0 < p \leq 1$, $0 < p_1 < p$ and $0 < p_2 < p$. Then, the contribution of $b[s]$ to the entropy $EH(b)$ of histogram b is $-p \log p$. The contribution of $b[s_1]$ and $b[s_2]$ to the entropy $EH(b')$ of histogram b' is $-p_1 \log p_1 - p_2 \log p_2$. The net increase $\delta EH(b, b')$ of entropy from $EH(b)$ to $EH(b')$ is given by the following (based on the property $x \log x < 0$ for all $0 < x < 1$).

$$\delta EH(b, b') = EH(b') - EH(b) = -p_1 \log p_1 - p_2 \log p_2 + p \log p = -p \left(\frac{p_1}{p} \log \frac{p_1}{p} + \frac{p_2}{p} \log \frac{p_2}{p} \right) > 0.$$

Therefore, $EH(b') > EH(b)$. □

Lemma 2. $JE_{LB}(I_1, I_2) = E(I_1)$ if $I_1 = I_2$.

Proof. If $I_1 = I_2$, then $JE(I_1, I_2) = E(I_1) = E(I_2)$ because the 2-D histogram B of I_1 and I_2 holds a one-to-one mapping to the 1-D histogram b_1 of I_1 (or, equivalently, to the 1-D histogram b_2 of I_2).

$$B[s_1][s_2] = \begin{cases} b_1[s_1] & \text{if } s_1 = s_2 \\ 0 & \text{if } s_1 \neq s_2 \end{cases} \text{ for all } 0 \leq s_1, s_2 < S.$$

□

Lemma 3. $JE_{LB}(I_g, I_G) \geq \max(E(I_g), E(I_G))$.

Proof. Since the 2-D joint histogram of two images has entropy at least as large as the entropy of individual 1-D histograms, the joint entropy of two images is at least as large as the individual entropies of the two images. □

Theorem 1. $JE_{LB}(I_1, I_2) = EH(B^*(I_1, I_2))$.

Proof. In the histogram b_1 of image I_1 , consider the element s_1 with the maximum value $c_1 = b_1[s_1]$, that is, $c_1 \geq b_1[s]$, $0 \leq s < S$. Similarly, in the histogram b_2 of image I_2 , let s_2 be the element with maximum value $c_2 = b_2[s_2]$. Without loss of generality, let $c_1 \leq c_2$ (otherwise, we can simply swap the identities of the images in the discussion). Let $\hat{I}_1^{-s_1}$ be a “trimmed” image obtained by omitting all c_1 occurrences of s_1 from I_1 . Let $\hat{I}_2^{-s_1}$ be the image obtained by omitting c_1 occurrences of s_2 in I_2 .

Let $\hat{e}(s_1) = JE_{LB}(\hat{I}_1^{-s_1}, \hat{I}_2^{-s_1})$ be the lower bound on the joint entropy of the trimmed images.

We claim that

$$\begin{aligned} JE_{LB}(I_1, I_2) &= e^* \\ \text{where } e^* &= \hat{e}(s_1) - p \log p, \\ p &= c_1/C, \\ \text{and } C &= \sum_{s=0}^S b_1[s]. \end{aligned} \tag{14}$$

Note that the shade s_2 is mapped to shade s_1 (and possibly to additional shades if $b_2[s_2] > b_1[s_1]$). Therefore, the contribution to the joint entropy from shade s_2 is $-p \log p + \alpha$ where $\alpha \geq 0$ is the contribution from mapping of $\delta s_{12} = b_2[s_2] - b_1[s_1]$ counts of shade s_2 to other shades.

The claim of lower bound in Equation 14 is proved as follows. Let us consider the value e^* provided by Equation 14. This value e^* is a valid candidate for the lower bound $JE_{LB}(I_1, I_2)$ because it is a feasible value for joint entropy. In other words, it provides a valid pair of images that have the same histograms as the original images but with a joint entropy equal to $e^* \geq JE_{LB}(I_1, I_2)$. If this candidate value happens to be the least possible value among all such pairs of images, then, it is indeed the lower bound value (replacing the inequality with equality, as $e^* = JE_{LB}(I_1, I_2)$). Hence, the problem at hand is to reason if it is possible to derive another different pairing of histograms (that is, another mapping of the images) that could result in a joint entropy lower than e^* . Let us suppose there is some other mapping that has a lower joint entropy $\hat{e} < e^*$. In that mapping of histograms, the element s_2 would be split across at least one more shade than in the candidate mapping; in other words, while the candidate mapping maps δs_{12} counts of s_2 to s_1 , any new mapping would split that mapping by at least one count. By Lemma 1, the entropy of the new mapping will be greater than the candidate entropy. Hence, it is not possible to improve upon the above candidate mapping that provides joint entropy equal to e^* . Therefore, $\hat{e}(s_1) = EH(B^*(\hat{I}_1^{-s_1}, \hat{I}_2^{-s_1}))$, $e^* = EH(B^*(I_1, I_2))$, and $JE_{LB}(I_1, I_2) = e^*$. \square

2. ALGORITHMS TO COMPUTE JOINT ENTROPY LOWERBOUND

Given two images, I_g and I_G , each of size $W \times H$ pixels, algorithms are presented in this section to compute a lower bound on the joint entropy between the two images. In a typical usage of joint entropy, the image I_G is extracted from a source image that is larger than I_g and from a potentially different spectrum.

Based on the notation, definitions, and relations developed in the preceding section, Algorithm 1 is designed to compute the lower bound of the joint entropy. An alternative expression of the main iterative loop of Algorithm 1 is developed in Algorithm 2 that uses fewer instructions. Finally, a heuristic of the same loop is developed in Algorithm 3 to estimate the lower bound. The heuristic reduces the run time at the cost of overestimating the lower bound on certain inputs.

The algorithms take as inputs two histograms b_g and b_G (corresponding to the images I_g and I_G , respectively) and produce a single real value which is the lower bound on the joint entropy between the two images. Algorithm 1 is a complete set of instructions. Algorithm 2 only lists the alternative loop and omits the parts prior to the loop which is identical with Algorithm 1. Similarly, Algorithm 3 lists only the modified loop that is different from Algorithm 1.

2.1 RUNTIME COMPLEXITY

The algorithms provide a non-trivial value for $JE_{LB}(I_g, I_G)$ by relying *only on the 1-D histograms of the individual images, and not relying on the exact joint entropy*. That is, given only the histograms b_{I_g} and b_{I_G} of the two images I_g and I_G , respectively, the algorithms compute $JE_{LB}(I_g, I_G) > \max(E(I_g), E(I_G))$, where $I_g \neq I_G$. The result is computed in $O(S \log S)$ time because it contains a sort operation on the histograms (which incurs a $O(S \log S)$ cost) followed by a linear pass on the non-zero values of the histograms (which incurs $O(S)$ time). The total time is constant because S is a small constant (typically, 256). Therefore, the algorithms are $O(1)$ (constant time) in practice.

2.2 MAIN ALGORITHM

Given the 1-D histograms b_g and b_G of images I_g and I_G , Algorithm 1 computes the entropy of the minimal 2-D joint histogram $B^*(I_1, I_2)$ defined in Equation 8. From Theorem 1, the computed entropy is equal to the non-trivial lower bound $JE_{LB}(I_g, I_G)$ of the two images.

The algorithm is a greedy iteration over the histograms. First, each histogram is sorted in descending order (lines 1-2). Then, the sorted histograms are traversed in lock-step fashion until the histograms are exhausted (line 4) or the first zero is encountered in either of the histograms (lines 5-7). When the first zero is encountered in the first histogram, it automatically signals the end of the other histogram also, because the total number of pixels is the same in the two images. This condition is asserted in line 6 before actually exiting the loop at line 7. While the maximum is non-zero, the loop continues processing.

Note that the entropy and joint entropy are invariant with a permutation of the shade identities. Due to this property, the sorted histograms may be seen as a loss-less remapping of the original images, and, consequently, histograms. For this reason, the shades can be safely referred to by the new indices after sorting even though they may be different from their identities before sorting. Therefore, in the following description, we will refer to the shades in the sorted histograms.

The two indices i_g and i_G keep track of the current location in each histogram respectively. The current maximum in b_g is always available at the element $b_g[i_g]$. Similarly, the current maximum in b_G is at $b_G[i_G]$. If $b_g[i_g] < b_G[i_G]$, then all the instances of shade i_g of I_g are mapped to the shade i_G of I_G . The assigned number of instances $b_g[i_g]$ is remembered in the variable c , which is later used in line 32 to update the joint entropy. The remaining count of instances of i_G is marked in $b_G[i_G]$ as equal to $b_G[i_G] - c$. This updated count should now be placed in its proper position to maintain the sorted order in the histogram b_G . This placement is accomplished by the **for** loop in lines 12-15, which swaps the values until sorted order is restored.

In the case of $b_g[i_g] > b_G[i_G]$, an analogous set of operations is performed in lines 17-25. In case $b_g[i_g] = b_G[i_G]$, the update is simpler: all instances of i_g are mapped to i_G , and both shades are exhausted

Algorithm 1: Computing the Joint Entropy Lowerbound

Result: Computes a lowerbound on the joint entropy of images I_g and I_G of same dimensions, given their histograms of color shades

Input

Data: S Number of bins of color shades
 C Number of pixels in I_g (same as in I_G)
 $b_g[0..S - 1]$ Histogram of I_g , $0 \leq b_g[\cdot] < C = \sum_s b_g[s]$
 $b_G[0..S - 1]$ Histogram of I_G , $0 \leq b_G[\cdot] < C = \sum_s b_G[s]$

Output

$JE_{LB}(I_g, I_G)$ Non-trivial lowerbound on the joint entropy of I_g and I_G

```

1 Sort  $b_g$  in descending order
2 Sort  $b_G$  in descending order
3  $JE_{LB} \leftarrow 0$ ;  $i_g \leftarrow 0$ ;  $i_G \leftarrow 0$ 
4 while  $i_g < S$  and  $i_G < S$  do
5   if  $b_g[i_g] = 0$  then
6     Assert  $b_G[i_G] = 0$ 
7     Exit loop
8   else if  $b_g[i_g] < b_G[i_G]$  then
9      $c \leftarrow b_g[i_g]$ 
10     $b_G[i_G] \leftarrow b_G[i_G] - c$ 
11     $b_g[i_g] \leftarrow 0$ 
12    for  $t_G = i_G$  to  $S - 1$  do
13      if  $b_G[t_G] \geq b_G[t_G + 1]$  then
14        break
15      swap( $b_G[t_G], b_G[t_G + 1]$ )
16     $i_g \leftarrow i_g + 1$ 
17   else if  $b_g[i_g] > b_G[i_G]$  then
18      $c \leftarrow b_G[i_G]$ 
19      $b_g[i_g] \leftarrow b_g[i_g] - c$ 
20      $b_G[i_G] \leftarrow 0$ 
21     for  $t_g = i_g$  to  $S - 1$  do
22       if  $b_g[t_g] \geq b_g[t_g + 1]$  then
23         break
24       swap( $b_g[t_g], b_g[t_g + 1]$ )
25      $i_G \leftarrow i_G + 1$ 
26   else  $b_g[i_g] = b_G[i_G]$ 
27      $c \leftarrow b_g[i_g]$ 
28      $b_g[i_g] \leftarrow 0$ 
29      $b_G[i_G] \leftarrow 0$ 
30      $i_g \leftarrow i_g + 1$ 
31      $i_G \leftarrow i_G + 1$ 
32    $d \leftarrow c/C$ 
33    $JE_{LB} \leftarrow JE_{LB} - d \log_2 d$ 

```

from both histograms, and hence, both histograms move on to the next most frequently occurring shades (lines 26-31).

In all cases, the index into the corresponding histogram (i_g for b_g , and i_G for b_G) is incremented if and only if the shade indicated by that index is entirely accounted for its contribution (by being set to zero) to the lower bound computation on line 33.

2.3 ALTERNATIVE LOOP OF THE MAIN ALGORITHM

Algorithm 2: A more compact but equivalent form of the loop in lines 4-33 of Algorithm 1	
1	while $i_g < S$ and $i_G < S$ do
2	if $b_g[i_g] = 0$ then
3	Assert $b_G[i_G] = 0$
4	Exit loop
5	$c \leftarrow b_G[i_G] - b_g[i_g] $
6	if $c = 0$ then $c \leftarrow b_G[i_G]$
7	$b_G[i_G] \leftarrow b_G[i_G] - c$
8	$b_g[i_g] \leftarrow b_g[i_g] - c$
9	if $b_G[i_G] = 0$ then $i_G \leftarrow i_G + 1$
10	else
11	for $t_G = i_G$ to $S - 1$ do
12	if $b_G[t_G] \geq b_G[t_G + 1]$ then
13	break
14	swap($b_G[t_G], b_G[t_G + 1]$)
15	if $b_g[i_g] = 0$ then $i_g \leftarrow i_g + 1$
16	else
17	for $t_g = i_g$ to $S - 1$ do
18	if $b_g[t_g] \geq b_g[t_g + 1]$ then
19	break
20	swap($b_g[t_g], b_g[t_g + 1]$)
21	$d \leftarrow c/C$
22	$JE_{LB} \leftarrow JE_{LB} - d \log_2 d$

Although Algorithm 1 is expressed for easier readability, it is possible to reduce the number of operations (code size). Algorithm 2 shows the same algorithm but rewritten so that the main loop has fewer instructions.

The key difference between Algorithm 1 and Algorithm 2 is the elimination of multi-way conditional spanning **if-else if-else if-else** into two independent, simplified **if-else** statements. The conditional checks about which of the two histograms has a larger maximum count than the other is eliminated. Also, the explicit check for equality of the maxima is also eliminated. Instead, the code is simplified to only depend on whether the current maximal shade is fully consumed in the mapping at any given iteration. This code

in Algorithm 2, which is exactly equivalent to that in Algorithm 1, may be useful in highly resource-constrained hardware implementations such as on Field Programmable Gate arrays (FPGAs), Application-Specific Integrated Circuits (ASICs), and Digital Signal Processing (DSP) chips.

2.4 HEURISTIC VARIANT OF THE MAIN ALGORITHM

Algorithm 3: A simplified heuristic version of the loop in lines 1-22 of Algorithm 2	
<pre> 1 while $i_g < S$ and $i_G < S$ do 2 if $b_g[i_g] = 0$ then 3 Assert $b_G[i_G] = 0$ 4 Exit loop 5 $c \leftarrow b_G[i_G] - b_g[i_g]$ 6 if $c = 0$ then $c \leftarrow b_G[i_G]$ 7 $b_G[i_G] \leftarrow b_G[i_G] - c$ 8 $b_g[i_g] \leftarrow b_g[i_g] - c$ 9 if $b_G[i_G] = 0$ then $i_G \leftarrow i_G + 1$ 10 if $b_g[i_g] = 0$ then $i_g \leftarrow i_g + 1$ 11 $d \leftarrow c/C$ 12 $JE_{LB} \leftarrow JE_{LB} - d \log_2 d$ </pre>	

The alternative expression of Algorithm 2 can be further simplified to Algorithm 3. This simplification is useful in reducing computational time, hardware resources and/or software code size even more. The simplification is achieved by eliminating the adjustment of partially assigned shades to restore the sorted order of the histograms when the maxima of both the histograms do not exactly match each other (in other words, when the maximum of one histogram is larger than that of the other). In that case, by eliminating the **for** loop that pushes the fractional remainder of the larger maximum to its proper position down its sorted histogram, two effects happen: (1) the run time is potentially reduced because the continual scan down the histogram array is eliminated, (2) the lower bound is potentially violated, resulting in a slightly higher estimate of the joint entropy than the theoretically correct value.

In applications where computational speed is more important at the cost of some tradeoff on accuracy, the abridged version given in Algorithm 3 can provide very fast execution and much smaller code size than Algorithm 1 and Algorithm 2.

3. CONCLUDING REMARKS

3.1 SOFTWARE IMPLEMENTATION

The algorithms have been implemented on central processing units (CPUs) as well as on graphical processing units (GPUs) in the C++ programming language and Common Unified Data Architecture (CUDA) programming environments. Software is available by contacting the authors.

3.2 POTENTIAL APPLICATION

Normalized Mutual Information (NMI) is a correlation metric that finds use in image registration and many other applications. NMI (Equation 11) is a function that relates the entropy measures of each individual image of the image pair with the joint entropy of the image pair. Although NMI offers excellent quality and resilience in matching, it is a computationally very intensive to apply in practice. The difficulty in actually computing NMI limits its use to small sizes of problems and non-real-time applications. At the crux of NMI computation is computing the joint entropy of the two images. A lower bound on the joint entropy places an upper bound on the NMI . The algorithms presented in this report are useful in improving the performance of traditional NMI -based image processing applications, especially in computing a non-trivial upperbound on $MNMI$ (Equation 13).

Based on the lower bound on joint entropy as presented in this report, an alternative to NMI can be formulated, called Normalized Mutual Information - Upper Bound (NMI_{UB} defined in Equation 12), which is a non-trivial upperbound on the actual NMI . Due to the higher speed of computation of JE_{LB} , NMI_{UB} can be computed faster than NMI while retaining (or exceeding) its matching quality and resilience. An empirical study of the effectiveness of NMI_{UB} based on JE_{LB} , in comparison to NMI in terms of runtime speed and quality of image matching, is being documented in a separate, complementary report [2].

4. REFERENCES

1. Mokshay Madiman and Prasad Tetali, "Information Inequalities for Joint Distributions, With Interpretations and Applications," in IEEE Transactions on Information Theory, vol. 56, no. 6, pp. 2699-2713, June 2010.
2. Kalyan S. Perumalla, Maksudul Alam, and Devin A. White, "Computational Speed and Matching Quality using an Upper Bound on the Normalized Mutual Information (NMI)," Technical Memorandum ORNL/TM-2017/87, Oak Ridge National Laboratory, March 2017.