Signal, Image and Video Processing (2020) 14:1293–1300
https://doi.org/10.1007/s11760-020-01668-6

ORIGINAL PAPER

Multi-focus image fusion for multiple images using adaptable size windows and parallel programming

Adan Garnica-Carrillo[1] · Felix Calderon[1] · Juan Flores[1]

Received: 18 September 2018 / Revised: 7 February 2020 / Accepted: 26 February 2020 / Published online: 10 March 2020

Abstract

The multi-focus image fusion with adaptable windows (MF-AW) algorithm for multiple images improves the results of the linear combination of images with variable windows (CLI-VV—from its Spanish acronym) algorithm, using a unique decision map and applying parallel programming. Other algorithms use the same window size throughout the image to produce a decision map; furthermore, a different decision map is produced for each pair of images. MF-AW determines the largest possible window size delimited by the edges of the decision map, which are improved using an iterative process. The execution time is improved using integral images, binary search, and parallel programming; as a result, the fused image is obtained in tenths of a second. Quantitative and qualitative measures indicate that the results obtained with this algorithm outperform the state of the art in terms of both accuracy and execution time.

Keywords Image fusion · Image segmentation · Adaptable size windows · Decision map

1 Introduction

In multiple applications, it is desirable to have the clearest possible digital images. However, the characteristics of camera lenses used in image capture systems have limitations that prevent completely clear images with all the details of objects at different distances. When we take a photograph, we decide which objects to focus the lens on, so the details are captured with greater clarity. Consequently, objects at different distances might appear sharper or blurrier. Kuthirummal et al. [1] define the depth of field as the range of distances from the camera lens (for a given aperture) that can be sharply reproduced. When an object is out of the range of distances covered by the depth of field, the object generates a colour circle that affects an area of the camera's sensor when it should only affect one point; those circles are known as blur circles [2]. The colours of the pixels in the generated image affected by blur circles result from a mixture of colours corresponding to several points of the scene, instead of the colour from a single point. This situation causes those pixels to appear blurred or unfocused.

Given a multi-focus image set $I = \{I(1), I(2), \ldots, I(N)\}$ of N images of size $n_r \times n_c$, each focused on objects at different distances, we define the multi-focus image fusion problem as the process of extracting the clear regions of each image $I(k)$ with the goal of forming a new image $J^\star$ that is as clear as possible. The fused image is more suitable for human perception or digital processing than are any of the images in the set. Henceforth, the subindices r, c are pixel coordinates that take values from the set $[1, \ldots, n_r] \times [1, \ldots, n_c]$.

To obtain the fused image with the greatest possible sharpness, it is necessary to have a measure of the sharpness of each image. Some works [3,4] propose spatial frequency to calculate the sharpness of a block or pixel.

MF-AW uses the absolute value of the convolution of the image with the Laplacian kernel [5–9] (shown in 1) as the sharpness function $F(k)$ given by 2 for the k-th image $I(k)$:

$$h = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{1}$$

$$F_{r,c}(k) = |h * I_{r,c}(k)| \tag{2}$$

✉ Adan Garnica-Carrillo
agarnica@dep.fie.umich.mx

Felix Calderon
calderon@umich.mx

Juan Flores
flores@umich.mx

[1] DEP-FIE, Universidad Michoacana de San Nicolás de Hidalgo, Morelia, Michoacán, Mexico

Since our goal is to produce the fusion of the images as efficiently as possible, we use the sharpness function 2 because convolution with a small kernel is faster than the discrete fast Fourier transform (convolution theorem, see [10]). This convolution operation can be seen as the sum of the partial derivatives over the 8 possible directions in the digital image.

This article is organized as follows: Sect. 1 describes multi-focus image sets and defines the image fusion problem. Section 2 describes the most relevant related work. Section 3 describes the MF-AW algorithm used to create a fused image from a set of multi-focus images. Section 4 provides details for its parallel implementation. Section 5 presents the results, and Sect. 6 presents the general conclusions.

2 State of the art

There exist many proposals in the literature to solve the multi-focus image fusion problem. Some are simple, producing a fused image based on the average of the images in the set. More elaborate proposals use artificial neural networks [11–14], differential evolution [15], discrete wavelet transform [3,16–18], contourlet transform [18,19], optimization processes [6,7,20], and parallel programming [21,22].

Other solutions assign a weight to each pixel of the images in the set and calculate the fused image as a weighted linear combination of the multi-focus images. In some cases, for pairs of images, the weights are binary and are called decision maps [8,9]. The fusion algorithms for multi-focus images can be divided into four groups according to the way images are processed: pixel-level processing [23–27], fixed-size block processing [7,15,20,28,29], using homogeneous regions [4,30], and using blocks or windows of variable size [8,31,32]. Each has its advantages and disadvantages. Pixel-level processing requires an additional process to eliminate inconsistencies in the decision map. Block-level processing has the disadvantage that it is necessary to decide the block size a priori. Consequently, if the selected block size is very large, the edge definition is lost; decreasing the block size makes the obtained result similar to the results obtained by pixel-level processing. In works where the process is done over homogeneous regions, a previous segmentation process is necessary to identify the regions. Calculating the block size adaptively solves the inherent problems found in fixed-size block and pixel-level processing; however, it introduces the problem of finding an appropriate block size. The most prominent and recent state-of-the-art work on multi-focus image fusion is discussed below.

Zhang et al. [29] decomposed images and extracted texture information to be used in a system of sliding windows to produce the fused image. However, the processing time required by the algorithm prevents its application in real time.

Farid et al. [33] developed an algorithm for multi-focus image fusion using content adaptive blurring (CAB) to detect sharper regions. They compute the fused image with a segmentation map initially set to the absolute difference between the original image and the CAB-blurred image. The segmentation map is refined using morphological operators and graph cut; it is then used to compute the fused image. Farid et al. compared their algorithm with fourteen other state-of-the-art algorithms, using the Lytro image dataset. Their implementation uses fixed-size blocks for all image pixels and is applied to pairs of images. In our opinion, the block size should not be the same for every image pixel.

Mansour et al. [9] proposed the use of a sparse dictionary to obtain the fused image. They compute the focus measure for each image extracting a focus feature for each pixel; then, a correlation process is applied using the dictionary to obtain the initial decision map. This decision map is processed using guided filtering and an MFR optimization process, producing a final decision map that is used to fuse the images.

Calderon et al. [8] presented the CLI-VV algorithm, which performs image fusion by maximizing a linear function and calculating a decision map. CLI-VV fuses a set of images with high accuracy in tenths of a second. The algorithm starts by fusing $I(1)$ and $I(2)$ to obtain $J(1, 2)$ and then $J(1, 2)$ and $I(3)$ are fused, and so on. By means of a counterexample, we have proven that merging a set of images by pairs is not entirely correct, since $I(1)$ and $I(2)$ are fused without taking into account the information of the remaining images $I(3), I(4) \cdots I(N)$.

Garnica et al. [20] computed a decision map using subsampling, scaling, and optimization over a sliding window system with fixed window size. Garnica *et al.* reported high accuracy; however, their execution times are greater than those reported in [8].

Based on the process presented in [8], MF-AW focuses on solving the fusion problem for a set of multi-focus images by calculating a single decision map. The decision map indicates which image contains the sharpest pixel at every location (r, c). The decision map $P_{r,c}$ allows us to extract the sharpest parts of each image and create a fused image.

3 Fusion with adaptable window size

Let us define the fused image $J(x)$ as a matrix of fused pixels, where each pixel is computed using a piecewise linear function with $N - 1$ linear equations (3),

$$J_{r,c}(x) = (\lceil x \rceil - x) I_{r,c}(\lfloor x \rfloor) + (x - \lfloor x \rfloor) I_{r,c}(\lceil x \rceil) \qquad 1 \leq x < N \tag{3}$$

where x is a real value that indicates how to combine the images to form a fused image. When $k \leq x \leq k + 1$, a fused pixel $J_{r,c}(x)$ with coordinates (r, c) combines information from only two pixels $I_{r,c}(k)$ and $I_{r,c}(k+1)$.

Instead of using a real variable x, an integer variable k is used to compute each fused pixel, where k is the image index from which the pixel will be taken (3). Therefore, the pixel $J_{r,c}(k)$ for $1 \leq k \leq N$ can be simplified to (4):

$$J_{r,c}(k) = I_{r,c}(k) \tag{4}$$

According to (4), there is a vector $\mathbf{J}_{r,c}$ that represents N pixels taken from their respective multi-focus image at coordinates r, c (5):

$$\mathbf{J}_{r,c} = [I_{r,c}(1), I_{r,c}(2), \ldots, I_{r,c}(N)] \tag{5}$$

The sharpness vector $\mathbf{F}_{r,c}$ given by (6) is computed via convolution of each image in the multi-focus set with the Laplacian kernel (2).

$$\begin{aligned}\mathbf{F}_{r,c} &= [|h * I_{r,c}(1)|, |h * I_{r,c}(2)|, \ldots, \\ &|h * I_{r,c}(N)|] \\ \mathbf{F}_{r,c} &= [F_{r,c}(1), F_{r,c}(2), \ldots, F_{r,c}(N)]\end{aligned} \tag{6}$$

We define the decision map $P_{r,c}$ as the index of the pixel in $\mathbf{J}_{r,c}$ with the greatest sharpness in $\mathbf{F}_{r,c}$, so $P_{r,c}$ is defined by (7):

$$P_{r,c} = \underset{k}{\operatorname{argmax}}\ F_{r,c}(k) \tag{7}$$

Our goal is to obtain a fused image $J^\star$ whose pixels $J^\star_{r,c} = I_{r,c}(P_{r,c})$ contain the maximum sharpness possible (pointed by $P_{r,c}$).

For the images shown in Fig. 1a–c by applying (7), we obtain the decision map P shown in Fig. 1d. Figure 1d shows, in black ($P_{r,c} = 1$), the pixels that are sharper in $I(1)$ than in $I(2)$ and $I(3)$, in grey ($P_{r,c} = 2$), the pixels of the scene that are sharper in $I(2)$ than in $I(1)$ and $I(3)$, and finally in white ($P_{r,c} = 3$) the pixels of the scene that are sharper in $I(3)$ than in $I(1)$ and $I(2)$. The values of the decision map are not always the same in regions corresponding to the same object. For example, it would be desirable for all pixels in the decision map that belong to the region that shows the sea lion in the foreground to appear in black; however, some points are grey or white. Therefore, some pixels corresponding to the sea lion will be taken from different images even though it is desirable for all of them to be taken from $I(1)$. We introduce spatial coherence to address this issue.

3.1 Spatial coherence

Spatial coherence means that regions that belong to the same object have similar sharpness. Spatial coherence in the decision map and consequently in the fused image can be achieved by adding constraints, as proposed by Calderon et al. [8] or using statistical measures such as the mean, median, or mode.

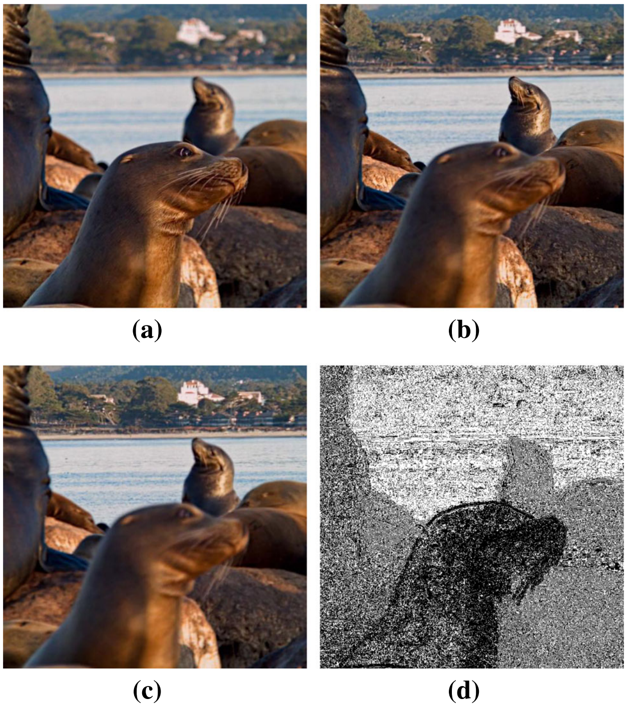

Fig. 1 Three multi-focus images of sea lions. **a**, **b** and **c** Multi-focus image set. **d** Decision map computed by (7)

In our approach, instead of solving a constrained optimization problem, we find the maximum of the average of the sharpness function for each image within a window of size $(2w+1) \times (2w+1)$ centred at each coordinate (r, c). The average sharpness $\bar{F}_{r,c}(k; w)$ for the k-th image is given by (8). Therefore, we construct a vector for a given window $\bar{\mathbf{F}}_{r,c}(w) = [\bar{F}_{r,c}(1; w), \bar{F}_{r,c}(2; w), \ldots, \bar{F}_{r,c}(N; w)]$, and the solution is the index of the element with the maximum value in the vector.

$$\bar{F}_{r,c}(k; w) = \frac{1}{(2w+1)^2} \sum_{l=r-w}^{r+w} \sum_{m=c-w}^{c+w} F_{l,m}(k) \tag{8}$$

For each image $I(k) \in I$, we compute the sharp function $F(k)$ and the integral image [34] of $F(k)$ as defined by (10). H can be computed in linear time with respect to the number of pixels in the images ($n_r \times n_c$).

$$\begin{aligned}H_{r,c}(k) = F_{r,c}(k) &+ H_{r,c-1}(k) \\ &+ H_{r-1,c}(k) - H_{r-1,c-1}(k)\end{aligned} \tag{9}$$

Based on the integral images H, which are computed once at the beginning of the process, $\bar{F}_{r,c}(k; w)$ can be determined

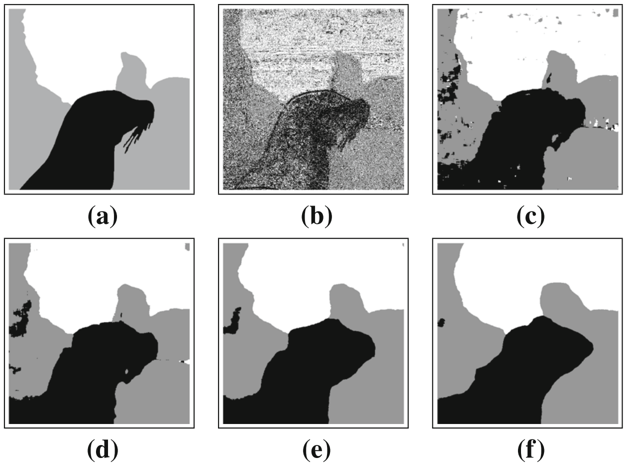

Fig. 2 Decision maps for the multi-focus image set in Fig. 1a–c. **a** Ground truth decision map. Decision maps for different w values **b** $w = 0$, **c** $w = 5$, **d** $w = 10$, **e** $w = 20$ and **f** $w = 40$

in constant time, independently of the window size, via (9).

$$\bar{F}_{r,c}(k; w) = \frac{1}{(2w+1)^2}[H_{r+w,c+w}(k) - H_{r+w,c-w-1}(k) - H_{r-w-1,c+w}(k) + H_{r-w-1,c-w-1}(k)] \tag{10}$$

Finally, the decision map at coordinates (r, c) $P_{r,c}$, given by (11), contains the index of the image pixel with the greatest average sharpness.

$$P_{r,c} = \underset{k}{\text{argmax}}\ \bar{F}_{r,c}(k; w) \tag{11}$$

The decision map P and the fused image J can be computed iteratively at each coordinate (r, c) in the image. The decision map at each coordinate $P_{r,c}$ can be computed using (11) and the fused image pixel $J_{r,c}(P_{r,c})$ using (3) for a given value w.

Figure 2 shows decision maps for the multi-focus image set in Fig. 1a–c. Figure 2a shows the ground truth decision map, and Fig. 2b–f shows the resulting decision maps computed using window sizes $w = 0$, $w = 5$, $w = 10$, $w = 20$, and $w = 40$, respectively. Note that the decision maps shown in Figs. 1d and 2b are equivalent, since they present no spatial coherence ($w = 0$). In addition, when w increases, the decision map loses border definition with respect to the multi-focus image set. The best results are obtained with an algorithm capable of automatically computing the optimal window size, providing the best trade-off between border definition and spatial coherence. In the following subsection, we describe an algorithm that achieves this goal.

3.2 Automatic calculation of window size

Different decision maps (thus, different fused images) can be obtained for different values of w. When w takes a value close to zero, good definition at the edges is obtained, but there is little spatial coherence; on the other hand, a large value for w produces greater spatial coherence at the expense of edge accuracy. The ideal situation would be to produce small windows near the edges (to achieve good segmentation) and large windows away from edges (to achieve spatial coherence in the decision map). This can be achieved by allowing the window size to vary and adapt to the image area where it is applied.

Calderon et al. [8] presented a complete analysis of the implications, advantages, and disadvantages of applying different values of w in segmentation/fusion and proposed a method for calculating the largest possible window restricted by the edges. A binary matrix B is defined with information obtained from the decision map edges. The value of $B_{r,c}$ is 1 if there is an edge in the decision map at location r, c and is 0 otherwise (12).

$$B_{r,c}(P_{r,c}) = \begin{cases} 1 & P_{r,c} \neq P_{r-1,c} \text{ or } P_{r,c} \neq P_{r,c-1} \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

Instead of using a unique window size w for all image pixels, we propose computing a different window size for each image pixel. $W_{r,c}$, the window size at location r, c, is computed by maximizing the area $A(W_{r,c}) = (2W_{r,c} + 1)^2$, limited by the number of edge pixels T within the window and a maximum window size $w_{\max}$, according to (13), assuming zeros in the image border.

$$\begin{aligned} W^{\star}_{r,c} = &\underset{W_{r,c}}{\text{argmax}}\ \ A(W_{r,c}) \\ &\text{s.t.} \\ &0 \leq W_{r,c} \leq w_{\max} \\ &\textstyle\sum_{l=r-w}^{r+w} \sum_{m=c-w}^{c+w} B_{l,m} < T \end{aligned} \tag{13}$$

The optimal window size $W^{\star}_{r,c}$ is computed by binary search (BS), and the sum of edge pixels in the window is calculated using integral images (9). The sum within each window is obtained with $(2w_{\max} + 1)^2$ operations. By contrast, when using incremental images, the sum is obtained with just 4 operations, so the reduction is $100 \times (1 - \frac{4}{(2w_{\max}+1)^2})\%$. For example, for $w_{\max} = 5$, the reduction is 96.69%. In contrast to Calderon et al., the approach proposed in this article computes the window size taking into account only one decision map for the image set instead of a decision map for each pair of images.

4 Parallel implementation

An image fusion decision map is computed using (11). The sharpness functions (2) for each of the images and the integral images are calculated in a previous process. Note that the calculation of these values $P_{r,c}$ and $W_{r,c}$ at coordinates (r, c) is totally independent of any other values at different coordinates. This allows us to make a parallel implementation, where each thread conducts the task of calculating the decision map values on a certain region of the image. We divide the rows by the number of available threads N_t to determine the region that each thread works on.

Splitting the image into regions enables parallel execution of the process. The idea is presented in Algorithm 1 MF-AW. Initially, the decision map $P_{r,c} = 1$ (line 5); consequently, in the first iteration $W^\star_{r,c} = w_{\max}$. Lines 8 to 14 of MF-AW calculate $W^\star$ and P iteratively for a maximum number of iterations N_{iter}. Line 10 loops to process each block assigned to each thread. (Each block contains $\mathcal{R}$ rows and n_c columns.)

```
Algorithm 1: MF-AW($I, T, w_{\max}, N_t, N_{\text{iter}}$)
1  Set $N \leftarrow |I|$ and $K \leftarrow \{1, 2, \ldots, N\}$;
2  Set $L \leftarrow [1, 2, \ldots, n_r] \times [1, 2, \ldots, n_c]$;
3  Compute $F_{r,c}(k)\ \forall k \in K, \forall (r, c) \in L$ using (2);
4  Compute $H_{r,c}(k)\ \forall k \in K, \forall (r, c) \in L$ using (9);
5  $P_{r,c} \leftarrow 1\ \forall (r, c) \in L$;
6  $\mathcal{R} \leftarrow \lceil n_r / N_t \rceil$;
7  for $t = 1$ to $N_{\text{iter}}$ do
8      Compute $B_{r,c}(P_{r,c})\ \forall (r, c) \in L$ using (12);
9      for $m = 1$ to $N_t$ (in parallel) do
10         for $r = (m-1)\mathcal{R}$ to $m\mathcal{R} - 1$ and $r < n_r$ do
11             for $c = 1$ to $n_c$ do
12                 Compute $W^\star_{r,c}$ using BS (13);
13                 Compute $\bar{F}_{r,c}(k; W^\star_{r,c})\ \forall k \in K$ (10);
14                 $P_{r,c} \leftarrow \operatorname{argmax}_{k} \bar{F}_{r,c}(k; W^\star_{r,c})$ (11);
15 Compute $J^\star_{r,c} \leftarrow I_{r,c}(P_{r,c})\ \forall (r, c) \in L$ using (4);
   Result: $J^\star$
```

The MF-AW and CLI-VV [8] algorithms are equivalent for two images. The main advantage of MF-AW is that it performs fusion with more than two images simultaneously, in contrast to the CLI-VV algorithm. The Results section focuses on illustrating the superior performance that MF-AW exhibits in time and accuracy.

5 Results

This section presents the results of MF-AW applied to different multi-focus image sets; the advantages of parallelizing the algorithm are highlighted. The experiments shown in this section were performed on a computer with a 64-bit Intel Core i7-7700HQ processor with 4 cores, 8 threads, and a 2.8 GHz frequency.

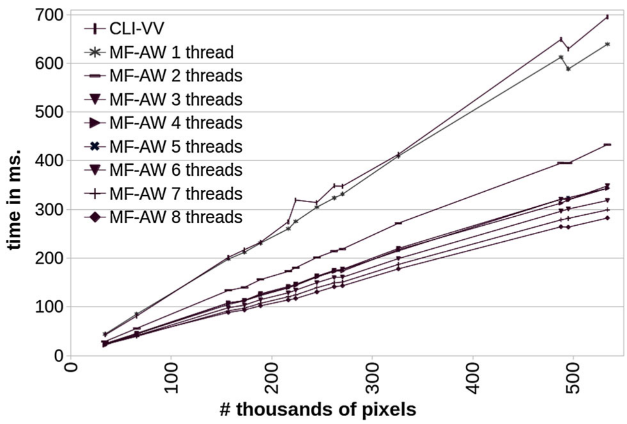

Fig. 3 Execution time for the CLI-VV and MF-AW algorithms with different image sizes for pairs of multi-focus images and multiple threads

The maximum window size was selected to create windows with a maximum area of 5% of the image area, $w_{\max} = \lceil (\sqrt{0.05 n_r n_c} - 1)/2 \rceil$ with a maximum number of pixel borders $T = 5$.

Calderon et al. [8] presented the CLI-VV algorithm and showed that it has linear time complexity. MF-AW can be shown to be equivalent to CLI-VV in the case of two multi-focus images. To show that MF-AW exhibits linear behaviour, we performed the fusion of different pairs of multi-focus images with different sizes, varying the number of threads, MF-AW preserves the linear behaviour, and the parallelization does not affect this characteristic. Figure 3 shows a plot of the execution time versus the number of pixels in the image: The plots in the figure show the behaviour of the CLI-VV and MF-AW algorithms with several threads. Note that linearity is preserved despite the parallel execution. In this figure, the upper line corresponds to CLI-VV: MF-AW outperforms CLI-VV, even when only one thread is used.

Table 1 shows the execution times for the CLI-VV and MF-AW algorithms for sets of multi-focus images of 375 × 500 pixels. In this table, the first column presents the multi-focus image set size N, columns 2 to 4 show the execution times (in ms) of CLI-VV and MF-AW with 1 and 8 threads, respectively, and columns 5 and 6 show the per cent time reduction (PTR) achieved by MF-AW. In this table, MF-AW with one thread has an execution time of 219 ms, outperforming CLI-VV with an execution time of 241 ms for a multi-focus image set with only two images. In addition, the time reduction becomes more evident as the image set size N grows; for example, the time reduction is 75.5% for a set size $N = 7$ when using one thread and 86.2% when using eight threads. In general, the time reduction is not provided only by parallelization but by the MF-AW algorithm itself.

Table 1 Execution time for the CLI-VV and MF-AW algorithms for images of size 375 × 500

N	Time (ms)			PTR (%)	
	CLI-VV	MF-AW		MF-AW	
		$N_t = 1$	$N_t = 8$	$N_t = 1$	$N_t = 8$
2	241	219	99	9.1	58.9
3	486	255	119	47.5	75.5
4	686	264	130	61.5	81.0
5	935	320	157	65.8	83.2
6	1137	326	166	71.3	85.4
7	1343	343	185	75.5	86.2

Table 2 Mean pixel accuracy results for 75 multi-focus image sets using the MF-AW and CLI-VV algorithms

N	Sets	Pixel accuracy			
		Fixed parameters		Optimum parameters	
		CLI-VV	MF-AW	CLI-VV	MF-AW
2	55	96.09	96.09	97.06	97.06
3	10	86.58	90.87	89.41	92.21
4	4	97.78	97.82	98.58	98.63
5	3	81.13	81.31	82.58	81.98
6	1	93.21	93.23	91.53	95.43
7	2	97.67	95.71	97.54	98.14

To test the algorithms' accuracy, a ground truth decision map $\tilde{P}$ was obtained by hand, trying to be as fair as possible with respect to objects that look the clearest in each of the multi-focus images. This decision map $\tilde{P}$ is compared with the decision maps P obtained by the CLI-VV and MF-AW algorithms with respect to the metric pixel accuracy (PA) [35]. PA counts all the pixels where the computed and reference bitmaps are different (14).

$$\mathrm{PA}(P, \tilde{P}) = \frac{100}{n_r n_c} \sum_{r=1}^{n_r} \sum_{c=1}^{n_c} \delta(P_{r,c} - \tilde{P}_{r,c}) \tag{14}$$

where $\delta()$ is 1 if the argument is 0 and is 0 otherwise.

Table 2 shows the PA for 75 multi-focus image sets with different sizes, N. The sets used are 55 pairs, 10 triads, 4 quartets, 3 quintets, 1 sextet, and 2 septets. Each row of Table 2 contains the mean PA for the set of images. Columns 3 and 4 show the PA computed with fixed parameters for both algorithms. Columns 5 and 6 show the results obtained with the best parameters ($w_{\max}$ and T) for each case. In most cases, MF-AW outperforms CLI-VV. In addition, PA is not affected by the number of images in the set, which affects only the execution time, as shown in Table 1.

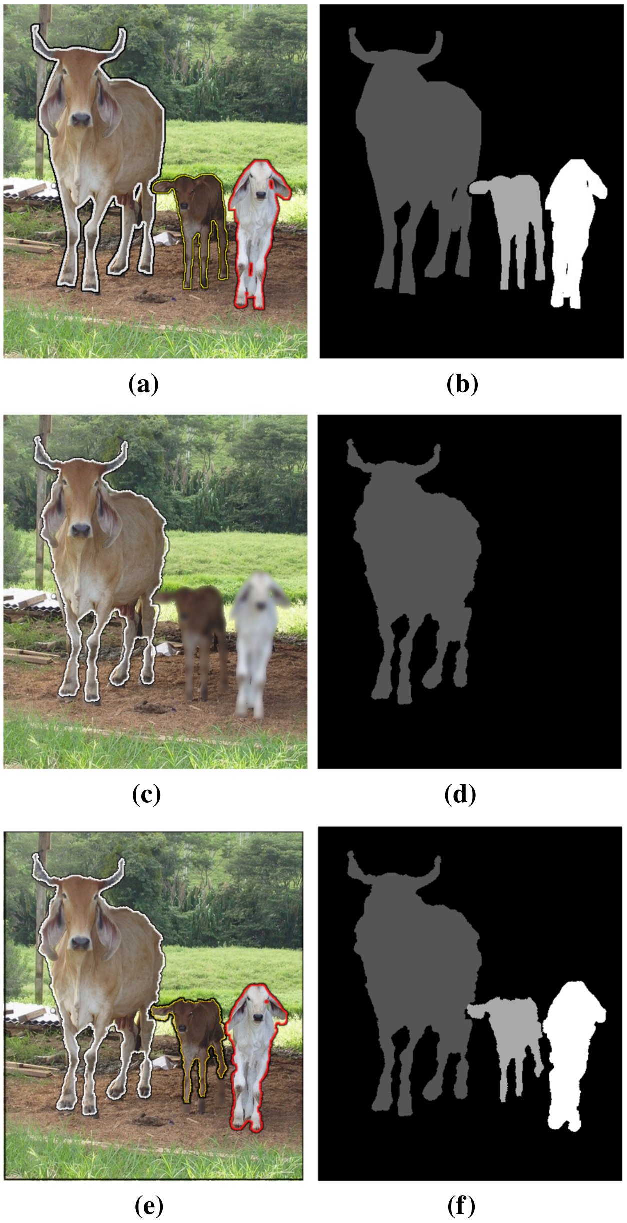

Fig. 4 Results for a multi-focus image set. **a** Fused image ground truth, **b** decision map ground truth, **c** fused image computed by CLI-VV, **d** decision map computed by CLI-VV, **e** fused image computed by MF-AW and **f** fused image computed by MF-AW

Figure 4 shows the resulting fused image for a multi-focus image set of size 4. The first column presents the fused image corresponding to a cow and its calves, with their respective decision map borders in the second column. Figure 4a shows the fused image computed with the ground truth decision map (Fig. 4b), Fig. 4c shows the fused image computed with the decision map obtained by CLI-VV (Fig. 4d), and Fig. 4e shows the fused image computed with the decision map produced by MF-AW (Fig. 4f). In this experiment, MF-AW outperforms CLI-VV with $PA = 98.6\%$ and $PA = 90\%$, respectively. Figure 4d shows an example where CLI-VV is more sensitive to parameter changes than is FM-AW; setting $w_{\max} = \lceil(\sqrt{0.1 n_r n_c} - 1)/2\rceil$ and $T = 5$ for both algorithms,

Table 3 Comparison of the normalized mutual information for the MF-AW algorithm and Mansour et al. algorithm [9]

Image	Mansour et al. [9]	MF-AW
Lab	1.26	1.16
Pepsi	1.25	1.32
Clock	1.25	1.20
Disk	1.15	1.19
Lytro dataset	1.19	1.21

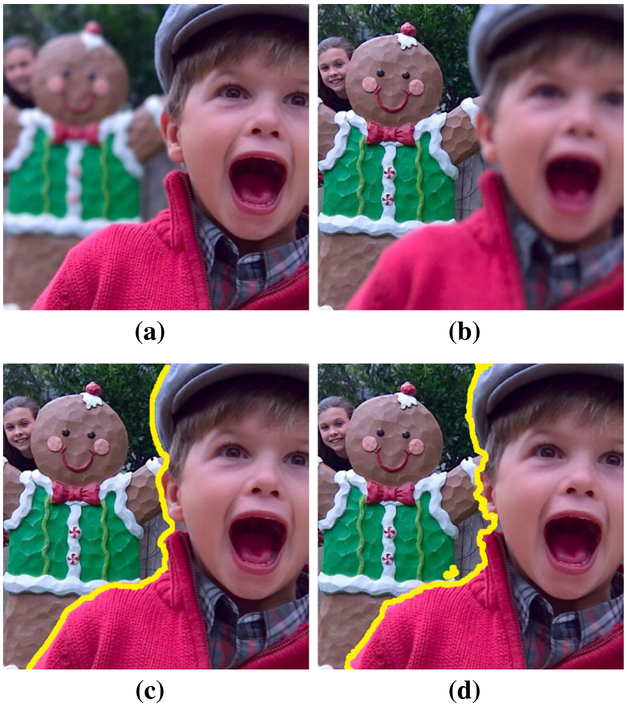

Fig. 5 Results for a multi-focus image set. Original images **a** $I(1)$ and **b** $I(2)$. Fused image resulting from **c** Mansour et al. [9] and **d** MF-AW

CLI-VV was unable to segment the two calves (that is why Fig. 4d is binary).

Table 3 presents a numerical comparison between MF-AW and the algorithm of Mansour et al. [9] (using the normalized mutual information (NMI) metric used in their publication) for four image sets and the Lytro dataset. According to the NMI metric, MF-AW is competitive with the algorithm of Mansour et al. Nevertheless, we believe the MF-AW algorithm is faster than the Mansour et al. algorithm due to its simplicity. We do not have an implementation to validate this hypothesis.

Table 4 shows the NMI and the execution time for the seven fastest algorithms presented in [33], CLI-VV and MF-AW applied to the Lytro dataset (image size is 520 × 520). Note that FM-AW shares the best NMI with the Farid *et al.*, PCNN, and CLI-VV algorithms. In addition, MF-AW has the second-best time. According to this table, MF-AW achieves the best trade-off between time and accuracy.

Figure 5a, b shows one multi-focus image set extracted from the Lytro data set. Figure 5c, d shows the fused images obtained with the decision map presented in [9] and MF-AW, respectively. Figure 5c, d shows, in yellow, the edge of the decision map. For this multi-focus image set, MF-AW obtains PA=99.1 and NMI=1.164 and the Mansour et al. algorithm obtains PA=98.8 and NMI=1.163. Both fused images look similar; nevertheless, MF-AW outperforms the Mansour et al. algorithm with respect to PA and NMI. In addition, MF-AW obtains the solution in less than one-tenth of a second, while [9] does not report the execution time.

6 Conclusions

We present a multi-focus image fusion algorithm MF-AW that exhibits execution times of tenths of a second. MF-AW behaves similarly to CLI-VV in the case of pairs of multi-focus images. According to the presented results, MF-AW exhibits better time performance and PA than CLI-VV.

MF-AW presents better precision than CLI-VV, since the latter performs the fusion of images by pairs, without taking into account the information of the other images. This fact causes the decision map to have information that biases the fusion procedure.

According to the results, MF-AW achieves results similar to those reported in [9] and [33]. However, the implementation of MF-AW is simpler and more efficient, producing the fused image in tenths of a second.

The parallelized MF-AW algorithm requires execution times approximately 90% shorter than those of CLI-VV for a set of seven multi-focus images. This implementation allows us to obtain the fused image in tenths of a second. Although the parallelization is not considered to be a scientific contribution, this article presents a simple algorithm in which the calculations of the decision map of any two pixels are independent. This independence enables parallelization. The simplicity of the algorithm could enable its direct imple-

Table 4 Comparison between the MF-AW algorithm and prominent state-of-the-art algorithms

	MWGF	Farid et al.	DCTV	PCNN	IFGD	CLI-VV	DCTLP	GIF	MF-AW	PCA
NMI	1.15	1.21	1.19	1.21	1.05	1.21	0.83	1.19	1.21	0.89
Time	4.07	1.87	1.71	1.45	0.84	0.35	0.30	0.25	0.14	0.03

mentation in camera hardware to allow the algorithm to be executed in real time.

References

1. Kuthirummal, S., Nagahara, H., Zhou, Changyin, Nayar, S.K.: Flexible depth of field photography. IEEE Trans. Pattern Anal. Mach. Intell. **33**(1), 58–71 (2011)
2. Sezan, M.I., Pavlovic, G., Tekalp, A.M., Erdem, A.T.: On modeling the focus blur in image restoration. In: 1991 International Conference on Acoustics, Speech, and Signal Processing, vol. 4, pp. 2485–2488. ICASSP-91 (1991)
3. Pajares, Gonzalo, de la Cruz, JesúManuel: A wavelet-based image fusion tutorial. Pattern Recognit. **37**(9), 1855–1872 (2004)
4. Li, Shutao, Yang, Bin: Multifocus image fusion using region segmentation and spatial frequency. Image Vis. Comput. **26**(7), 971–979 (2008)
5. Riaz, M., Park, S., Ahmad, M.B., Rasheed, W., Park, J.: Generalized Laplacian as focus measure. In: Bubak, M., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) Computational Science ICCS 2008. Lecture Notes in Computer Science, vol. 5101, pp. 1013–1021. Springer, Berlin (2008)
6. Calderon, F., Garnica, A.: Multi focus image fusion based on linear combination of images. In: 2014 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC), pp. 1–7. IEEE (2014)
7. Calderon, Felix, Garnica-Carrillo, Adan, Flores, Juan J.: Fusión de imágenes multi foco basado en la combinación lineal de imágenes utilizando imágenes incrementales. Revista Iberoamericana de Automática e Informática Industrial RIAI **13**(4), 450–461 (2016)
8. Calderon, F., Garnica-Carrillo, A., Flores, J.J.: Fusión de imágenes multi-foco con ventanas variables. Revista Iberoamericana de Automática e Informática industrial **15**, 262–276 (2017)
9. Nejati, Mansour, Samavi, Shadrokh, Shirani, Shahram: Multi-focus image fusion using dictionary-based sparse representation. Inf. Fusion **25**, 72–84 (2015)
10. Alan, V.O., Alan, S.W., Hamid, S., Nawab, S.H.: Signals and Systems. ISBN-10. Pearson Press, London (1996). ISBN 10: 1-292-02590-5
11. Amin-Naji, Mostafa, Aghagolzadeh, Ali, Ezoji, Mehdi: Ensemble of cnn for multi-focus image fusion. Inf. Fusion **51**, 201–214 (2019)
12. Ma, Y., Zhan, K., Wang, Z.: Applications of Pulse-Coupled Neural Networks. Higher Education Press, Beijing (2011)
13. Pagidimarry, Madhavi, Babu, K.Ashok: An all approach for multi-focus image fusion using neural network. Artif. Intell. Syst. Mach. Learn. **3**(12), 732–739 (2011)
14. Yang, Yong, Que, Yue, Huang, Shu-Ying, Lin, Pan: Technique for multi-focus image fusion based on fuzzy-adaptive pulse-coupled neural network. Signal, Image Video Process. **11**(3), 439–446 (2017)
15. Aslantas, V., Kurban, R.: Fusion of multi-focus images using differential evolution algorithm. Expert Syst. Appl. **37**(12), 8861–8870 (2010)
16. Zhang, Z., Blum, R.S.: A categorization of multiscale-decomposition-based image fusion schemes with a performance study for a digital camera application. Proc. IEEE **87**(8), 1315–1326 (1999)
17. Yang, Y.: A novel DWT based multi-focus image fusion method. Proc. Eng. **24**, 177–181 (2011). (**International Conference on Advances in Engineering 2011**)
18. Shah, Parul, Merchant, Shabbir N., Desai, Uday B.: Multifocus and multispectral image fusion based on pixel significance using multiresolution decomposition. Signal, Image Video Process. **7**(1), 95–109 (2013)
19. Yang, Yong, Tong, Song, Huang, Shuying, Lin, Pan: Multifocus image fusion based on nsct and focused area detection. IEEE Sens. J. **15**(5), 2824–2838 (2015)
20. Garnica-Carrillo, A., Calderon, F., Flores, J.: Multi-focus image fusion by local optimization over sliding windows. Signal, Image Video Process. **12**, 869–876 (2018)
21. Liu, Yu., Chen, Xun, Peng, Hu, Wang, Zengfu: Multi-focus image fusion with a deep convolutional neural network. Inf. Fusion **36**, 191–207 (2017)
22. Bejinariu, S.I., Rotaru, F., Nita, C.D., Luca, R.: Parallel approach for multifocus image fusion. In: 2013 International Symposium on Signals, Circuits and Systems (ISSCS), 1–4 July (2013)
23. Lewis, J.J., O'Callaghan, R.J., Nikolov, S.G., Bull, D.R., Canagarajah, N.: Pixel- and region-based image fusion with complex wavelets. Inf. Fusion **8**(2), 119–130 (2007). (**Special Issue on Image Fusion: Advances in the State of the Art**)
24. Aslantas, Veysel, Toprak, Ahmet Nusret: A pixel based multi-focus image fusion method. Opt. Commun. **332**, 350–358 (2014)
25. Li, Shutao, Kang, Xudong, Fang, Leyuan, Jianwen, Hu, Yin, Haitao: Pixel-level image fusion: a survey of the state of the art. Inf. Fusion **33**, 100–112 (2017)
26. Kumar, B .K.Shreyamsha: Multifocus and multispectral image fusion based on pixel significance using discrete cosine harmonic wavelet transform. Signal, Image Video Process. **7**(6), 1125–1143 (2013)
27. Shreyamsha Kumar, B.K.: Image fusion based on pixel significance using cross bilateral filter. Signal, Image Video Process. **9**(5), 1193–1204 (2015)
28. Yin, W., Zhao, W., You, D., Wang, D.: Local binary pattern metric-based multi-focus image fusion. Opt. Laser Technol. **110**, 62–68 (2019). (**Special Issue: Optical Imaging for Extreme Environment**)
29. Zhang, Yongxin, Chen, Li, Zhao, Zhihua, Jia, Jian: Multi-focus image fusion based on cartoon-texture image decomposition. Optik. Int. J. Light Electron Opt. **127**(3), 1291–1296 (2016)
30. Pramanik, Sourav, Prusty, Swagatika, Bhattacharjee, Debotosh, Bhunre, Piyush Kanti: A region-to-pixel based multi-sensor image fusion. Proc. Technol. **10**, 654–662 (2013)
31. Bai, X., Zhang, Y., Zhou, F., Xue, B.: Quadtree-based multi-focus image fusion using a weighted focus-measure. Inf. Fusion **22**, 105–118 (2015)
32. De, I., Chanda, B.: Multi-focus image fusion using a morphology-based focus measure in a quad-tree structure. Inf. Fusion **14**(2), 136–146 (2013)
33. Farid, M.S., Mahmood, A., Al-Maadeed, S.A.: Multi-focus image fusion using content adaptive blurring. Inf. Fusion **45**, 96–112 (2019)
34. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, vol. 1, pp. I–511–I–518 (2001)
35. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. CoRR arXiv:1411.4038 (2014)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.