An Evaluation Mechanism for Saliency Functions Used in Localized Image Fusion Quality Metrics

M. Hossny, S. Nahavandi and D. Creighton
Centre for Intelligent Systems Research
Deakin University
Australia
{mhossny, nahavand, dougc}@deakin.edu.au

Abstract— Image fusion is a proven value adding technique for image analysis. Automated image fusion aims to give the fusion system the ability to select, analyze and evaluate fusion-worthy images. This paper examines the evolution of present techniques used for assessing quality of image fusion operators. It also presents an algorithm that objectively evaluates the realism of saliency functions used in image fusion quality measures. Most image fusion quality metrics depend on estimating the amount of information transferred from each source image into the fused image. This algorithm rebuilds the fused image using the estimated information from each source image and compares it to the original fused image.

Keywords—Image Fusion, Quality Metrics.

I. INTRODUCTION

The process of image fusion aims to merge two or more images to produce a new image that is better than the original ones. The term ‘better’ differs from one context to another. In some contexts, it means holding more information. In other contexts, it means getting more accurate result or reading. In general, an image fusion system takes as an input two or more source images and produces one fused image as an output. The fusion process applies a fusion operator repeatedly on
the source images and/or intermediate output images. As a matter of fact, almost all the present image fusion operators are binary ones. The need for ternary or higher narity fusion operator has not yet been identified.

Fusion theory was originally founded to allow multi-opinion decision-making. It can be defined as a typical application of Dampster-Sheffer theory. Afterwards, researchers developed several signal-, pixel- and feature-level fusion techniques [24] and [25]. Researchers have developed several definitions of image fusion. A formal definition of image fusion was derived by Wald [3]. Other definitions can be found in [4] and [5]. As a part of the definition, a fusion operator must maintain the closure property which means its output must be of the same kind as its inputs.

Judging fusion algorithms objectively is not an easy task. It is even harder than objectively judging quality for single images being processed. Wang discussed difficulties regarding objective image quality assessment in [11]. Assessing the quality of an image fusion operator depends mainly on the amount of information transferred from each source image into the fused one. It also depends on how clear the transferred information is. An expressive objective quality assessment technique allows a certain level of autonomy in the process of image fusion. It also extends image fusion definition so it can fit to several applications. A fusion system that is able to identify informativity of images prior to fusion is also capable of estimating how much more information was added after fusion regardless of the content of the image. Such a fusion system can also manage the amount of information transferred from each source image.

A wide variety of applications can make use of objective quality assessment. Automated battlefield, augmented reality, remote sensing and surveillance are all applications of image fusion. Even some artistic effects, like texture transfer [23], can be considered as image fusion algorithms. For example, in an automated battlefield where a swarm of robots are gathering information from a sensor network or directly from the field, there must be a scale defining how good or bad the captured images are before sending them. If a robot is running in a dark environment, a visual image
will actually give nothing, while a thermal one will be far more informative. However, the darkness itself is a valuable piece of information. Such information allows the robot to allocate the proper resources, namely bandwidth, to transmit both images. It also can embed the estimated quality in the image prior sending, as Wang described in [13]. This will allow the receiver to enhance the received image to reach the hidden quality message.

Xydeas and Petrovic [12] estimated fusion performance based on edges in the image. Zhang and Blum [7] used a mixture of Rayleigh probability density functions to model image histogram and estimate quality of noisy images. Mutual information measure was examined, using joint histogram, by Qu [10] and Hossny et al. in [28, 29]. It described the use of joint histogram between a fused image with each of the source images. Local cross-correlation of the feature maps of the source and fused images was studied by Zhao in [14]. Buntlov and Bretschneider [8] applied multi-level thresholding to variance maps in order to identify the spatial blocks holding more information and, probably, should be transferred into the fused image. They concluded that quality measures of image fusion algorithm should be extended to take into considerations segmented regions and weight averaging their contribution in assessment of quality based on their areas and how much information each region holds. They have derived a segmentation-based solution in [17].

An excellent survey about quality measures was presented by Blum [15] and Hossny et al. in [30]. Most of the described techniques assume a single-pass image fusion system with two source images and rely on measuring how far the resulting fused image is from both source images. The overall metric measures how much information was transferred from source images to the fused image. Piella [2] added a saliency factor for each pair of corresponding blocks (a block from each input image) being examined against the corresponding block in the fused image. In this paper an algorithm for objective evaluation of how realistic image fusion quality metrics are.

The rest of this paper is organized as follows. Section 2 presents the evolution quality metrics used to assess image fusion quality. Section 3 lists saliency functions and discusses their advantages and
limitations. Section 4 describes the proposed algorithm that measures how realistic these assessment methods are. Experiments, results and analysis are derived in Section 5. Finally, Section 6 concludes and introduces to future advancements.

II. EVOLUTION OF IMAGE FUSION QUALITY METRICS

Quality metrics for image fusion algorithms has evolved from image quality metrics. This section follows the evolution journey of image quality metrics into image fusion quality metrics. It starts with listing image quality metrics followed by its realization in the field of image fusion. Localization implementation of these quality metrics as well as their weighted version is presented next.

Throughout this paper, two-image (binary) fusion operator is referred to as \( f_o \) and a set of fusion operators is referred to as \( F_0 \) and \(|F_0|\) is number of fusion operators being examined. The quality metric is referred to as \( QM \) and is written as \( QM(x,y|f) \) in equations, where \( x \) and \( y \) are source images and \( f \) is the fused image. Quality difference between two images is written as \( Q_0(x,y) \), where \( x \) and \( y \) are the images being examined. Kernel window parameters are written as \( w \), while the set of windows being used is written as \( W \) and \(|W|\) is the number of windows.

Since the image processing field began, researchers have developed several techniques of quality assessment starting with standard deviation and statistical measures of dispersion, entropy and histogram based methods, MSE and RMSE and signal to noise ratios. Then they were followed by edge detection based methods and hybrid algorithms and frequency domain methods. Finally it ended with UQI and SSIM in [1] and [16]. There is no ultimate quality metric that satisfies all the cases and avoids all problems. SSIM and UQI methods have been proved to be capturing the structural similarity between the two images being tested but it still has room for improvement. A comparative study of image quality metrics is beyond the scope of this paper.

The realization of these methods into image fusion quality metrics is quite simple. The main idea is to measure the amount of information transferred from source images into the resulting fused image.
Simple averaging of quality difference between fused image and each of the source images participated in the fusion process will do the job.

\[ Q_M(x, y|f) = \frac{Q_0(x, f) + Q_0(y, f)}{2} \]

The general form for \( N \) source images will then be;

\[ Q_M(x_1, ..., x_N|f) = \frac{1}{N} \sum_{i=1}^{N} Q_0(x_i, f) \]

where \( x_i | i = 1, ..., N \) are source images in a multi image fusion process.

Cross entropy, information measure and universal quality index are examples of image quality assessment. This realization depends mainly on the used quality metric, its tuned parameters and how sensitive it is to information presented in images. For example, standard deviation is very sensitive to noise. Entropy and cross-entropy are strong against noise but sometimes lose track of new changes. MSE and RMSE are sensitive to \( L_2 \) errors. Blum has presented a very good comparison for these metrics for night vision experiments in [15].

Localization of these methods is a very important improvement. It runs the quality metric via a convolution-like moving window scheme. Source images and the fused image are subdivided into simple overlapping blocks. Quality is estimated for corresponding blocks. In this case the quality metric will be;

\[ Q(x, y) = \frac{1}{|W|} \sum_{w \in W} Q_0(x, y|w) \]

where \( Q_0(x, y|w) \) is the quality metric applied to a window \( w \), \( W \) is the set of all windows and \( |W| \) denotes to number of windows. The quality of fusion algorithm is then estimated by;
The main benefit of localization of quality metric is to capture structural information encoded in the image. Selecting the best window size is not easy. It differs according to the used quality metric and the amount of information presented in images and its distribution. In general, large window size drives the quality metric one step backwards to a non-localized version. On the other hand selecting a tiny window size loses the information shared with neighbor blocks and causes instabilities for some metrics like standard deviation and histogram-based methods. The reason behind these instabilities is that tiny windows tend to have same color sub-images leading to zero-valued saliency blocks.

As a matter of fact, source images contribute differently in the fusion process. Some regions add more information in the fused image. Therefore, these blocks should have higher effect on the quality metric than other less-informative blocks. In [2] Piella and Heijman have introduced a weighted version of the localized image fusion quality metrics and defined it as;

\[
QM(x, y|f) = \frac{1}{|W|} \sum_{w \in W} \frac{Q_0(x, f|w) + Q_0(y, f|w)}{2}
\]

where \( \lambda_s, w \) is the weighting factor and is defined as;

\[
\lambda^{s,w} = \frac{s(x|w)}{s(x|w) + s(y|w)}
\]

\[
\tilde{\lambda}^{s,w} = 1 - \lambda^{s,w}
\]

where \( s(x|w) \) and \( s(x|w) \) are local saliencies of both input images and \( \lambda^{s,w} \) has a dynamic range of [0, 1]. This definition of weighting factors takes into consideration the amount of information in source images and identifies blocks’ contributions based on a linear mapping. However, the fused
image changes according changes in fusion operators and fusion parameters. This weighting factor does not take into consideration the amount of information presented in the fused image. Therefore, the weighting factor should be defined differently:

$$\lambda^s,w = \frac{s(x,f|w)}{s(x,f|w) + s(y,f|w)}$$

The weighting factor is computed in a way that favors the blocks and regions in the source image that have more impact in the fused image. Local saliency is, simply, an arbitrary feature that estimates the amount of information in each region in source and the fused images. Standard deviations, dynamic range of colors, or entropy are examples of the features used as saliency functions [2]. Quadtree decomposition was used in [27]. Other non-differentiable saliencies have been presented [6] and [26].

Another saliency-depending weighting factor was added in [2]. It affects the way the overall quality is calculated. It favors blocks and regions rather than averaging with number of windows $|W|$. The overall quality will then be:

$$QM(x,y|f) = \sum_{w \in W} c_w \left( \lambda^{s,w}Q_0(x,f|w) + \hat{\lambda}^{s,w}Q_0(y,f|w) \right)$$

$$c_w = \frac{C_w}{\sum_{w' \in W} C_w}$$

where $c_w$ represents the perceptual importance of each block in the process of the overall quality estimation. Piella and Heijman chose this factor to be:

$$C_w = \max(s(x,f|w), s(y, f|w))$$
III. Saliency Functions

In this section, a survey about different saliency functions is presented. For each function, advantages and limitations are discussed. Throughout this section, saliency maps \( s(x, f|w) \) are derived for source images presented in Fig. 1 and are colored using Jet color map. Contribution maps \( \lambda^{s,w} \) are also derived for the same source images and coloured using a gray scale color map.

Fig. 1: Source images

\textit{a. Standard Deviation:} It has been used for decades to estimate the amount of information stored in images. Simply a standard deviation kernel is applied to each of the source images. It captures the actual information distribution in the image to the finest detail (Fig. 1-left). However, it does not detect gradient change in the second image (Fig. 1-right).

\[
\sigma_x = \sqrt{\frac{\sum_{i=1}^{N} (x_i - \mu_x)^2}{N - 1}}
\]

where \( x_i \) is the sampled colors in the image \( x \), \( \mu_x \) is the average color and \( N \) represents the number of samples.
b. Entropy: Entropy is a histogram-based measure that estimates number of bits needed to represent a finite digital image. It depends mainly on assigning the least number of bits to the most frequently appearing color.

c. Dynamic Range: This feature is the simplest to compute. It estimates the amount of information by the difference of the maximum and the minimum color value in the entire image or block. Although it is not a non-differentiable function, it produces overlapping block shaped regions in the contribution map as shown in Fig. 4. This is because of contiguous blocks sharing same features and therefore same dynamic range. This may affect the overall quality assessment.

Fig. 3 shows how it captures both noisy detail in first images and the color gradient in the second image. The entropy function is defined as follows;
Fig. 3: Top: Entropy maps for source images presented in Fig.1. Bottom: λ-maps for same images.

Fig. 4: Top: Dynamic range maps for source images presented in Fig.1. Bottom: λ-maps for same images.
\[ e = -\sum_{i=1}^{G} p_i \log_2 p_i \]

where \( G \) is the number of colors in the image and \( p_i \) represents the percentage of using the color \( i \) in the image.

d. **Quad-Tree Decomposition**: It is a technique that, recursively, subdivides square image into a set of images according to some criterion. Dynamic range, entropy and standard deviation are examples of criteria. However, they suffer from same problems discussed before as saliency functions. More details about quadratic tree decomposition algorithms, data structures and applications are presented in [20-22]. This saliency function has been presented in [15] as an improvement of universal quality index.

This saliency function estimates the amount of information transferred into the fused image by calculating the change of block sizes between each source image and the fused one. When a large block in first image becomes smaller in the fused image this means that a new piece of information from the second source image have been added and vice versa. Fig. 5 shows quadratic tree decomposition maps of both source images.

\[ \lambda_w = \begin{cases} 
1 - \frac{d_f - d_x}{d_y - d_x} & \text{if } d_x \neq d_y \\
0.5 & \text{otherwise}
\end{cases} \]

where \( d_x, d_y \) and \( d_f \) are the dimensions of the corresponding blocks in source images \( x \) and \( y \) and the fused image \( f \). This saliency function is totally non-differentiable. This leads to sharp-edged blocks in saliency maps. Low pass filters for the saliency maps can help reducing the effect of sharp-edged blocks.

The main benefit of this saliency function is that it involves the fused image in calculating the
saliency map. This allows the quality metric using this saliency function to operate and give consistent realistic measures for all fusion operators.

e. Covariance: In [6] Cvejic proposed to estimate saliency according to the covariance between each of source images and the fused image. Therefore, this saliency function changes contribution maps according to the used fusion algorithm. The contribution map is then defined as:

\[
\lambda_{s,w} = \begin{cases} 
0 & \text{if } \frac{\sigma_{xf}}{\sigma_{xf} + \sigma_{yf}} < 0 \\
\frac{\sigma_{xf}}{\sigma_{xf} + \sigma_{yf}} & \text{if } 0 \leq \frac{\sigma_{xf}}{\sigma_{xf} + \sigma_{yf}} \leq 1 \\
1 & \text{if } \frac{\sigma_{xf}}{\sigma_{xf} + \sigma_{yf}} > 1 
\end{cases}
\]

This equation is non-differentiable which may lead to sharp-edged blocks in the contribution maps unless the kernel is applied with overlapping like in Fig. 6.
IV. PROPOSED ALGORITHM

This section presents the proposed algorithm to identify how realistic different saliency functions are. In this section, the algorithm assumes binary fusion operators and examines weighted localized quality metrics. The algorithm listed in Fig. 7 works on two source images $x$ and $y$, one fused image $f$, a saliency function $s$ and a kernel window $w$. As shown in Fig. 8, it calculates the saliency maps for both source images, selects the most contributing parts from them and builds a new fused image.
Finally, it returns the mean-squared-distance between the built fused image with the original one $f$. This error indicates how accurate the examined saliency function selects the best parts from source images. The reason for using MSE as a global measure is that it is the oldest standard measure. In order to ensure accurate judging of saliency functions, this algorithm should be applied to fused images resulting from all identified fusion operators using same source images and kernel window.

Mean $\mu_{FO}^{s,w}$ and variance $\sigma_{FO}^{s,w}$ of the resulting errors describe the expected accuracy of the saliency function $s$ and how stable the saliency function $s$ is, respectively. The subscript $FO$ represents the set of fusion operators against which the saliency measure $s$ is being examined with a kernel size $w$.

---

Algorithm saliency_metric

Inputs: $s$, $x$, $y$, $f$, and $w$

Output: $e_{s,w}$

1. Calculate saliency map for both source images with a window size of $w$
   
   $x_{smap} \leftarrow s(x, f|w)$
   
   $y_{smap} \leftarrow s(y, f|w)$

2. Calculate $\lambda$-weight map
   
   $\lambda^{s,w} = \frac{x_{smap}}{x_{smap} + y_{smap}}$

   $\hat{\lambda}^{s,w} = 1 - \lambda^{s,w}$

3. Get the most contributing parts from source images
   
   $x_{selected} \leftarrow \lambda^{s,w} \times x$

   $y_{selected} \leftarrow \hat{\lambda}^{s,w} \times y$

4. Build an estimated fused image with the most contributing parts in source images.
   
   $f_{new} \leftarrow x_{selected} + y_{selected}$

5. Calculate estimated distance of the new fused image and the original one.
   
   $e_{s,w} \leftarrow MSE(f, f_{new})$

6. Return $e_{s,w}$

---

Fig. 7: Pseudo code listing for the proposed algorithm.
\[ \lambda = \frac{s(x, f|w)}{s(x, f|w) + s(y, f|w)} \]

Fig. 8: Workflow diagram of the proposed algorithm
\[
\mu_{FO}^{SW} = \sum_{f_0 \in FO} \frac{Q_0(f_0(x,y), \lambda_{f_0}^{SW} \cdot x + \lambda_{f_0}^{SW} \cdot y)}{|FO|}
\]

\[
\sigma_{FO}^{SW} = \sum_{f_0 \in FO} \frac{\left( Q_0(f_0(x,y), \lambda_{f_0}^{SW} \cdot x + \lambda_{f_0}^{SW} \cdot y) - \mu_{FO}^{SW} \right)^2}{|FO|}
\]

\[
\lambda_{f_0}^{SW} = \frac{s(x, f|w)}{s(x, f|w) + s(y, f|w)}
\]

where \( f = f_0(x,y) \) is the fused image.

A saliency function depending on standard deviation of source images will not give consistent results with other fusion algorithms. This is because saliency maps remain the same, while the fused image changes according to the used fusion operator.

On the other hand, Cvejic’s and QTD saliency function are expected to give more consistent results since it includes the fused image in computing saliency maps. In fact both \( \mu_{FO}^{SW} \) and \( \sigma_{FO}^{SW} \) are very important and present a valuable interpretation of the saliency function being examined. Therefore, an arbitrary constant \( \alpha \) is used to tune the algorithm and the overall error would then be;

\[
E_{FO}^{SW,\alpha} = \mu_{FO}^{SW\alpha} \cdot \sigma_{FO}^{SW1-\alpha}
\]

V. EXPERIMENTS AND SENSITIVITY ANALYSIS

In this section experiments, results and sensitivity analysis are discussed. The aim of the experiment discussed next is to compare between different saliency functions discussed earlier in this paper. The experiment simply runs the algorithm presented in section four on the source images presented in Fig. 1. Both source images are 256×256 pixels.

The experiment tests saliency functions against a set of known fusion operators with different window sizes. The experiment calculates average error resulting from using saliency functions when used with different fusion operators like selecting minimum or maximum, PCA, DWT, as well as contrast, ratio, gradient, Laplacian, morphological and FSD pyramids. Fig. 9 and Fig. 10 compare
between different saliency functions. It shows how the mean and variance of errors change as the window size increases.

Variance driven quadtree decomposition records the least average error for all compared image fusion operators. However, it does not seem to be giving consistent results with different fusion operators as shown in Fig. 9. On the other hand, Cvejic’s saliency function presents higher average error but with more stability across different fusion operators. The main reason behind that is the fact that they allow the fused image to participate in calculating the $\lambda$-weight. Fig. 11 and Fig. 12 compare the behavior of saliency functions with different $\alpha$-factors.

![Graph showing comparison between saliency functions](image)

Fig. 9: A comparison between saliency functions discussed in this paper. It presents window size along the x-axis and $\mu_{F0}^{SW}$ along the y-axis.
VI. CONCLUSIONS AND FUTURE WORK

In this paper localized image fusion quality metrics are presented. It discusses the evolution of image fusion quality metrics starting from image quality metrics, their application in image fusion and ending with a weighted localized version. A new algorithm for estimating how realistic and accurate localized image fusion quality metrics are is also presented. The paper also compares between different weighting functions. It discusses how realistic they are and the reasons behind their advantages and limitations.

Three improvements should be added in future. First, a database of benchmark problems for image fusion should be built. It should cover declared problems in different fields using image fusion. Second, a set of axioms for designing an image fusion quality metric should be declared. With these
axioms one can guarantee that new saliency functions fit with the fusion operators. For example, [9], [18] and [19] present a new different trend of estimation of information distribution among the images being examined. They employ foveation to identify the regions attracting the most of human attention. Third, mean $\mu_{FO}^x$ and variance $\sigma_{FO}^x$ along the window size should be added to the equation calculating the overall error.

![Graph](image)

Fig. 11: A comparison between saliency functions discussed in this paper. It presents window size along the x-axis and $E_{FO}^{s,w,\alpha}$ along the y-axis with $\alpha = 0.75$ favoring $\mu_{FO}^{s,w}$. 
Fig. 12: A comparison between saliency functions discussed in this paper. It presents window size along the x-axis and $E_{F0}^{sw,\alpha}$ along the y-axis with $\alpha = 0.25$ favoring $a_{F0}^{sw}$.

REFERENCES


