CONSISTENT IMAGE PROCESSING BASED ON CO-SALIENCY

XIANGNAN REN1,2 | JINJIANG LI1,2 | ZHEN HUA3,2 | XINBO JIANG1,4

1School of Computer Science and Technology, Shandong Technology and Business University, Yantai, China
2Co-innovation Center of Shandong Colleges and Universities, Future Intelligent Computing, Shandong Technology and Business University, Yantai, China
3School of Information and Electronic Engineering, Shandong Technology and Business University, Yantai, China
4Shandong Provincial Key Laboratory of Software Engineering, Shandong University, Jinan, China

Abstract
In a group of images, the recurrent foreground objects are considered as the key objects in the group of images. In co-saliency detection, these are described as common saliency objects. The aim is to be able to naturally guide the user's gaze to these common salient objects. By guiding the user's gaze, users can easily find these common saliency objects without interference from other information. Therefore, a method is proposed for reducing user visual attention based on co-saliency detection. Through the co-saliency detection algorithm and matting algorithm for image preprocessing, the exact position of non-common saliency objects (called Region of Interest here, i.e. ROI) in the image group can be obtained. In the attention retargeting algorithm, the internal features of the image to adjust the saliency of the ROI areas are considered. In the HSV colour space, the three components H, S, and I are adjusted separately. First, the hue distribution is constructed by the Dirac kernel function, and then the most similar hue distribution to the surrounding environment is selected as the best hue distribution of ROI areas. The S and I components can be set as the contrast difference between ROI areas and surrounding background areas according to the user's demands. Experimental results show that this method effectively reduces the ROI areas' attraction to the user's visual attention. Moreover, comparing this method with other methods, the saliency adjustment effect achieved is much better, and the processed image is more natural.

1 | INTRODUCTION

With the rapid development of the information age, the amount of video and picture data is increasing exponentially. With the huge amount of image information, people hope that their computer can quickly locate the key information in an image, and so image saliency detection technology has developed rapidly and has been widely used in the fields of image target detection, image segmentation, and video or image compression. However, the information contained in a single image is limited. In contrast, the image group contains richer information. Therefore, some researchers have proposed the co-saliency detection algorithm. In the image group, the recurrent foreground objects are considered as the key objects in the group of images. In co-saliency detection, these are called common saliency objects. The saliency map obtained by the co-saliency detection algorithm is still a greyscale image. The larger the greyscale value of a pixel, the stronger the saliency of the point. As with a single image, it is hoped that subsequent processing of the image group will be performed according to the main content of the image group to improve computational efficiency and reduce the computational burden of the computer.

In recent years, the co-saliency detection algorithm has been gradually improved and widely used in various fields. For example, co-segmentation [1] uses co-saliency detection technology to find common salient targets in related image groups and segment them; in video and image compression technology [2], co-saliency detection technology is used to determine the compression rate of different areas of the video or the image in...
the image group. Subsequently, only the areas with high co-saliency are losslessly compressed, and lossy compression is performed on the low-saliency or non-common salient areas so that the compression coding can be efficiently obtained and less storage space is occupied.

Co-saliency objects should meet the following conditions: firstly, all co-saliency objects should exist in each image of the image group; secondly, co-saliency objects should be the foreground area in the images; and thirdly, the co-saliency objects should have saliency features in all images. Figure 1 shows a typical example of co-saliency detection, in which only the players in the red strip are the co-saliency objects in the image set, and the other information is all non-common saliency objects.

Information can be guided to modify the salient region in the image, so that the co-saliency objects in the image group have unique saliency in each image, to guide human visual attention. This can be applied to many practical scenes, such as smart advertising, image editing, object enhancement, and improving image aesthetics, etc. According to the saliency guidance information, the image features in the area of the original image are randomly changed to achieve manipulation of the area saliency in the image. This study is called “attention retargeting”. Imagine that consumers can see products at first glance in flyers or magazines, so that attention to the products will be greatly improved; in the works of amateur photographers, non-important objects are often more conspicuous in position or colour than important objects. Using the image attention retargeting algorithm based on co-saliency can better correct the aesthetic effects of the image, enhance the objects, or eliminate the interference objects.

However, the saliency map obtained by the saliency detection algorithm is sparse pixel points and cannot be used as the guide information for subsequent image feature processing. That is to say, in image attention retargeting, the saliency estimation and image feature editing need two separate steps, but it is not enough to find the feature clues only by the saliency detection algorithm, the specific location of the area to be edited also needs to be found. Therefore, the matting algorithm is used to obtain non-common salient object areas, that is, image blocks guiding saliency (hereinafter referred to as ROI areas). The cluster-based co-saliency detection algorithm [3] can be applied to both single and multiple images. By comparing the single and co-saliency maps, it can discover which areas are non-common saliency objects clearly, and then use the KNNT (K-Nearest Neighbours Texture) matting technology [4] to extract non-common saliency objects accurately and completely. The image attention retargeting algorithm proposed can process multiple areas in the image simultaneously, and edit the image features based on the HSI colour space according to the consistent processing method of the internal features of the image, to achieve non-common saliency objects closer to the characteristics of the background area and to reduce the saliency.

Section 2 summarises previous related work. Section 3 introduces in detail the preprocessing methods used and the new image processing methods proposed. In Section 4, qualitative analysis and quantitative analysis of the experimental results of this method are carried out, as well as comparative experiments to prove the effectiveness of this method. Finally, conclusions and a summary of possible future improvements are presented in Section 5.

2 | RELATED WORK

2.1 | Co-saliency detection algorithm

The co-saliency detection algorithm is an emerging branch of saliency detection. In 2010, Chen et al. [5] first proposed the image co-saliency detection algorithm. The traditional co-saliency detection algorithm research [6–9], by dividing the input image into several small computing units, such as super-pixel block [10] or pixel cluster [3], can effectively simulate the corresponding relationship between images in an image group. In 2013, Tan et al. [11] proposed a self-contained co-saliency detection algorithm based on the super-pixel affinity matrix.

However, the object of these algorithms proposed at first was only for image pairs. In 2014, Liu et al. [12] proposed an image co-saliency detection algorithm based on hierarchical segmentation. The algorithm is divided into two modules: fine segmentation and coarse segmentation. Region similarity and region contrast measurement are based on fine segmentation. Based on the coarse segmentation, the prior objects of each area are measured based on the connectivity of the image boundaries. Finally, based on the region similarity measure, the global similarity of each area is obtained, and then it is effectively integrated with the internal saliency map and the prior object map to generate a co-saliency map for each image. Li et al. [13] proposed a new co-saliency detection model based on graph matching. The model integrates visual appearance, saliency coherence, and spatial structure continuity for co-saliency detection between image pairs. Zhang et al. [14]
proposed a hierarchical image co-saliency detection framework from rough to fine to capture features. This method generates the co-saliency map through a mask-guided fully convolutional network.

In recent years, some researchers have extended deep learning techniques to co-saliency detection algorithms and obtained good performance by co-training common saliency objects [15–21]. Zhang et al. [15–17] proposed a self-paced multiple-instance learning framework for co-saliency detection [17]. On the one hand, it is self-paced through MIL (Multiple-Instance Learning) can adapt to insight measurement and discover common patterns in areas of co-saliency. On the other hand, it tends to ensure the reliability and stability of learning by simulating human learning processes. Tsai et al. [18] proposed a phased co-saliency detection algorithm. In the first stage, they evaluated the quality of salient features through an unsupervised deep learning model with an SAE (Stacked AutoEncoder). In the second stage, they developed a self-training convolutional neural network to alleviate the problem of excessive smoothing of saliency maps.

Gao et al. [19] proposed an FCN (Fully Convolutional Network) framework embedded with a common attention module, called Co-Attention FCN (CA-FCN). This method inserts the common attention module into the high-level convolutional layer of FCN. This can assign a larger attention weight to common salient objects and a smaller attention authority to the background and non-common interferers to improve the final saliency detection performance. Although the development of the co-saliency detection algorithm has only been in the last few years, it has gained a high level of attention and very satisfactory performance and has shown a broad application prospect.

2.2 Matting algorithm

In 1984, Porter et al. [22] made a mathematical definition for the matting problem for the first time, expressing any observation image \( I \) as a linear combination of the foreground image \( F \) and the background image \( B \) under the \( \alpha \) channel, as shown in (1), where \( \alpha \) represents the opacity of the foreground object. Since the foreground image \( F \), background image \( B \) and \( \alpha \) value are unknown, the expression of alpha matting is ambiguous.

\[
I = \alpha F + (1 - \alpha)B, \quad \alpha \in [0, 1] \tag{1}
\]

In the past decade, natural image matting has received great attention from the research community, and can now be considered as one of the classic research problems in visual computing. Natural image matting algorithms can be roughly divided into three categories:

1. Sampling-based matting algorithms. This method has the foreground and background information of each unknown pixel estimated from the surrounding known pixels, so that the alpha value can be calculated by finding the optimal foreground and background data pair in the known area. Ruzon and Tomasi [23] proposed a sampling-based matting method to separate the background and foreground by establishing a colour space model. This method can extract objects from natural images and extend the blue screen matting to images with almost any colour background. Then, some scholars found that the sampling-based matting algorithm has problems of conflicts between multiple sampling standards and incomplete sample space, and made optimisation improvements [24,25]. The PDMS method (Pixel-level Discrete Multi-object Sampling) proposed by Huang et al. [24] formalises the colour sampling process at each unknown pixel as a multi-objective optimisation problem and expands the sample space. This greatly improves the boundary clarity of the results.

2. Propagation-based matting algorithm. In 2004, Sun et al. [26] proposed a Poisson matting algorithm. This method assumes that the foreground and background are locally smooth. According to the input trimap, Dirichlet boundary conditions are used to solve the Poisson equation to reconstruct the mask value of the unknown region. However, because pixels cannot be solved independently, this method is not as good as the sampling method in parallelism. Wang et al. [27] proposed a robust matting algorithm. This method greatly increases the sampling range and optimises the mask value by adding sampling coefficients. In 2013, Chen et al. [28] applied non-local principles to general Alpha matting to extract multiple image layers simultaneously. This algorithm improves the non-local matting by considering the first \( K \) nearest neighbours in the high-dimensional feature space, called KNN Matting. Aksoy et al. [29] proposed a natural image matting algorithm based on pure affinity. The algorithm uses multiple definitions of pixel affinity to control the flow of information from known opaque areas and within the location area itself.

3. Learning-based matting algorithm. In recent years, deep learning has been widely used and has achieved good results in natural image matting [30–34]. Cho et al. [30] proposed the DCNN (Deep Convolutional Neural Networks) matting algorithm, which for the first time introduced a deep neural network to the image matting technology. This method uses the network to learn the results of different methods, to achieve the fusion of local and non-local methods. Xu et al. [31] proposed a large-scale data set neural network model, which mainly solves the problem of using only low-level features and lacking high-level context semantic information. Tang et al. [34] proposed a method based on a mixture of sampling and learning. Before opacity estimation, a deep neural network was used to estimate the layer colour, which greatly improved the performance of opacity estimation.
2.3 | Image attention retargeting algorithm

One of the earliest results of image attention retargeting research was a technique proposed by Su et al. [35] to selectively alter texture changes to reduce image regional significance. The visual attention retargeting methods have been roughly divided into two categories: colour-based methods [36–39] and direction-based methods [40,41].

The colour-based method modifies each colour component so that the visual saliency inside the ROI increases, while the visual salience outside the ROI decreases. The advantage of this method is that it not only increases the significance of the ROI area but also maintains the high resolution of the non-ROI. Hagihara et al. [36] proposed an iterative technique for manipulating image saliency. This method adjusts the colour and brightness of the image (inside and outside the ROI) in each step after selecting the ROI to increase saliency. During this period, the saliency map of the modified image will be calculated and analysed, and adjusted to increase the saliency within the ROI until it is considered the highest relative to the rest of the image. However, because saliency must be calculated in each step, this method has high computational complexity. Mechrez et al. [39] proposed a method for manipulating object saliency by considering the internal colour and saliency of the image. The optimisation framework in this method uses patches in the same image to modify the saliency of the ROI area.

On the other hand, the direction-based method guides the user’s visual attention to the non-blurred area by blurring the colours outside the specified area. Hata et al. proposed a direction-based image modification method [40], which created a modified image by using a Gaussian filter to blur areas outside the specified area. In this method, high-frequency components are suppressed by applying spatial filtering to control the resolution. Hitomi et al. [41] proposed a saliency mapping based on wavelet transform and an image modification method. This method corrects the frequency components according to the obtained saliency map to suppress the visual saliency outside the specified area.

Recently, Chen et al. [42] proposed a model based on deep learning, which directly optimises operations between any distinguishable saliency estimation method and neural networks that apply image processing to meet saliency guidance, called the SaGIM (Saliency-Guidance Image Manipulation) model.

3 | METHODS PROPOSED

An image attention retargeting algorithm was studied. In the image preprocessing stage, the authors adopted the existing KNNT matting algorithm [4] and cluster-based co-saliency detection algorithm [3]. In this section, the two algorithms used are introduced and the image attention retargeting algorithm proposed is described in detail. Figure 2 is the main flow chart of the method. First, the authors used the saliency detection algorithm for each input image to obtain a single-image saliency map, and then the co-saliency detection algorithm was used on the image group to obtain a co-saliency map. Second, by comparing two saliency maps, a non-common salient object was obtained (i.e., ROI), and the matting algorithm used to extract the corresponding positions of the non-common salient objects. Finally, the attention of the ROI area was retargeted to achieve the effect of weakening its saliency.
3.1 | Image preprocessing

3.1.1 | Co-saliency detection algorithm

In the saliency map extraction stage, a cluster-based co-saliency detection algorithm was selected [3]. This method uses clustering to maintain global consistency among multiple images and fused bottom-up clues to generate the final co-saliency map. This method does not require deep learning and is simple, versatile and efficient.

The algorithm is based on a two-layer clustering method to perform co-saliency detection of multiple images. The algorithm flow is shown in Figure 3. Given a set of images, this method uses K-means clustering to group the pixels in a single image and cluster the pixels in multiple images to form associations among multiple images. This method selects uniqueness, distance from the centre of the image, and repetability as the salient features on a single image and multiple images, respectively, called contrast cue, spatial cue, and corresponding cue. Finally, a probability framework is used to soft-assign each pixel based on all clustering characteristics to compute the saliency value, which is used to generate the final saliency map.

Symbol description: The pixel with index m in image \( M^i \) is represented as \( \{ p_m^i \}_{m=1}^{L_i} \), where \( L_i \) represents the i-th image lattice. \( \{ r_m^i \}_{m=1}^{L_i} \) represents the normalised position of pixel \( p_m^i \) in image \( M^i \). Clustering N images \( \{ M^i \}_{i=1}^{N} \), we get K clusters \( \{ C_k^i \}_{k=1}^{K} \). The clusters are represented by a series of D-dimensional vectors \( \{ d_k^i \}_{k=1}^{K} \) and \( d_k^i \) represents the cluster centre of cluster \( C_k^i \). Define a mapping function \( f(p_m^i) \) for pixel \( p_m^i \) and cluster index \( f(p_m^i): \mathbb{R}^2 \rightarrow \{ 1 \ldots K \} \).

Contrast Cue: The contrast cue represents the uniqueness of visual features on a single image or multiple images. This feature can better represent the global correspondence than the pixel level. The contrast cue \( c^c(k) \) of cluster \( C_k^i \) is defined by the contrast with all other clusters, as shown in Equation (2).

\[
    c^c(k) = \sum_{i=1, i \neq k}^{K} \left( \frac{n_i}{N} \| d_k^i - d_k^i \|_2 \right) 
\]

where \( n' \) represents the number of pixels in cluster \( C \), and \( N \) represents the number of pixels in all images.

Spatial Cue: In the human visual system, the closer the object is to the centre of the image, the more attention it will attract. According to the “central bias rule” in saliency detection, the distance between the object and the image centre increases, and the attention gain decreases. We define the spatial cue \( c^s(k) \) of cluster \( C_k^i \), as shown in Equation(3).

\[
    c^s(k) = \frac{1}{n_k^i} \sum_{p=1}^{N} \sum_{q=1}^{L_p} \left[ K \left( \left\| \theta_{pq}^i - 0 \| \right\| 0, \sigma^2 \right) \cdot \delta \left[ f(p_q^i) - C_k^i \right] \right] 
\]
where $\delta(\cdot)$ is the Kronecker delta function, $\alpha^B$ represents the centre of the image $M^B$, here the Euclidean distance between the pixel $r_i^B$ and the image centre $\alpha^B$ is calculated using the Gaussian kernel $K(\cdot)$, the variance $\sigma^2$ is the normalised radius of the image, and the normalised coefficient $n^k$ is the number of pixels in the cluster $C^k$.

**Corresponding Cue:** The corresponding cue is used to measure the cluster distribution on multiple images. It is an important attribute of co-saliency to describe the repeatability of an object frequently appearing in multiple images. We use clustering to roughly estimate the distribution range of clusters in multiple images.

First, an $N$-bin histogram $\hat{h}_i^k = \{\hat{h}_i^k\}_{i=1}^N$ is used to describe the distribution of cluster $C^k$ in $N$ images, as shown in Equation (4).

$$\hat{h}_i^k = \frac{1}{n^k} \sum_{j=1}^{L^k} \delta[f(p_j^i) - C^k], \quad i = 1 \ldots N \tag{4}$$

where $n^k$ is the number of pixels in cluster $C^k$, and $\hat{h}_i^k$ satisfies the condition $\sum_{i=1}^{N} \hat{h}_i^k = 1$. Then, we define the corresponding cue as Equation (5).

$$c'(k) = \frac{1}{\text{var}(\hat{h}_i^k) + 1} \tag{5}$$

where $\text{var}(\hat{h}_i^k)$ is the variance of the histogram $\hat{h}_i^k$ of cluster $C^k$. The higher $c'(k)$ within a cluster, the more uniform the pixel distribution in each image.

According to the above process, the contrast cue and spatial cue of a single image are obtained, as well as the contrast cue, spatial cue and corresponding cue of multiple images. For the single-image saliency map, it is obtained by fusing its two cues. For the multi-image saliency extraction stage, the three cues and the single-image saliency map are fused, and finally, the multi-image co-saliency map is obtained. Common fusion methods include linear summation and point-wise multiplication. The point-by-point multiplication is more advantageous in suppressing noise, and the linear summation has a better recall rate. In the algorithm experiment, a saliency map with higher accuracy is needed. Therefore, point-wise multiplication is selected in this algorithm to integrate the final saliency features.

Before merging saliency features, all cluster feature distributions need to be standardised to standard Gaussian distributions. First, the co-saliency probability of cluster $C^k$ as $P(C^k)$ is defined, as shown in Equation (6).

$$P(C^k) = \prod_i c_i^p(k) \tag{6}$$

where $c_i^p(k)$ is the saliency cues.

So far, the distribution of the cluster-level co-saliency values obtained is discrete and needs to be smoothed. The saliency likelihood probability $P(p|C^k)$ of the pixel $p$ in the cluster $C^k$ satisfies the Gaussian distribution $N$, as shown in Equation (7).

$$P(p|C^k) = N\left(\left\|V_r \cdot d^k_p\right\|_2^0, \sigma_k^2\right) \tag{7}$$

where $V_r$ is the feature vector of pixel $p$ and the variance of cluster $C^k$ is regarded as Gaussian variance. Finally, the saliency probability of each pixel is obtained by summing the co-saliency probabilities on the cluster.

$$P(p) = \sum_{k=1}^{K} P(p, C^k) = \sum_{k=1}^{K} P(p|C^k)P(C^k) \tag{8}$$

### 3.1.2 | KNNT matting algorithm

In Equation (1), the observed image is represented as a linear combination of the foreground $F$ and the background $B$ under the $\alpha$ channel. The value of $\alpha$ is $[0,1]$. When the value of $\alpha$ is 1, the pixel belongs to the foreground area. When the value of $\alpha$ is 0, the pixel belongs to the background area. When the value of $\alpha$ is between $\langle 0,1 \rangle$, the pixel belongs to the unknown area of image $I$. Therefore, how to estimate the value of the unknown area by using the pixels of the known foreground and background areas is the key to the matting problem. In areas where the foreground and background have overlapping colours or dense textures, the quality of the estimated mask value will be greatly affected. The authors use the KNNT matting technology [4]. This adds texture features based on KNN matting to enhance the background constraint, and the experimental effect is better.

**Non-local Principle [43]:** Suppose a denoising pixel $p$ is a weighted sum of pixels with similar characteristics and a kernel function $\kappa(i,j)$, as follows.

$$E[X(i)] \approx \sum_j X(j) \kappa(i,j) \frac{1}{D_i} \tag{9}$$

$$\kappa(i,j) = \exp\left(-\frac{1}{b_1^2} \|X(i) - X(j)\|_2^2 - \frac{1}{b_2^2} \|d_j\|^2\right) \tag{10}$$

$$D_i = \sum_j \kappa(i,j) \tag{11}$$

where $X(i)$ is the feature vector of pixel $i$, $d_j$ is the distance between pixels $i$ and $j$, $\| \cdot \|_2$ is a Gaussian norm, $b_1$ and $b_2$ are constants. Similar to Equation (9), the expected value of $\alpha$ is as follows.
\[ E[\alpha_i] \approx \sum_j a_i \kappa(i,j) \frac{1}{D_j} \quad \text{or} \quad D_i \alpha_i \approx \kappa(i, \cdot) \alpha \quad (12) \]

where \( \alpha \) is a vector of all \( \alpha \) values of the input image. The non-local principle also applies to Equation (12). When the given distribution \( E[\alpha_i|X(i) = X(j)] = \alpha_i \) holds, pixels with the same characteristics have the same value of \( \alpha \). All pixels in the input image are represented as a matrix \( D\alpha \approx A\alpha \), where \( A \) is the similarity matrix of \( N \times N \), \( D \) is a diagonal matrix of \( N \times N \), and \( N \) is the total number of pixels in the input image.

That is, \((D - A)\alpha \approx 0\) or \(\alpha^T L \alpha \approx 0\), where \(L = (D - A)^T(D - A)\) is a clustered Laplace operator, which resolves the quadratic minimisation problem \(\min_{\alpha} \sum A_{ij} (\alpha_i - \alpha_j)^2\).

Feature Texture Calculation: Three vectors are defined \(V_L = (1, 4, 6, 4, 1)\), \(V_E = (-1, -2, 0, 2, 1)\) and \(V_S = (-1, 0, 2, 0, -1)\), respectively, represent the line feature operators, edge feature operators and point feature operators, and calculate four templates \(V_L^T \times V_E\), \(V_E^T \times V_L\), \(V_L^T \times V_S\), and \(V_S^T \times V_L\). The linear filter response of the template at point \((i, j)\) in image \(I\) is \(t_{ij} = t_{ij}^L \times V_L(i, j)\), \(t_{ij}^E \times V_L(i, j)\), \(t_{ij}^S \times V_L(i, j)\), and \(t_{ij}^E \times V_L(i, j)\). And the texture information at this point is shown as Equation (13).

\[
t(i, j) = \left[ \left(t_{ij}^L \times V_L(i, j) \right)^2 + \left(t_{ij}^E \times V_L(i, j) \right)^2 \right]^{1/2} \quad (13)
\]

Then normalising Equation (13), Equation (14) is obtained.

\[
T(i, j) = \frac{t(i, j)}{t_{\text{max}}} \quad (14)
\]

where \(t_{\text{max}} = \max\{|t(i, j)|, 0 \leq i \leq M - 1, 0 \leq j \leq N - 1\}\), \(M\) and \(N\) are the height and width of image \(I\) respectively.

Feature Vector Calculation: The texture feature is added to the KNN matting method, so in the HSV colour space, the feature vector of a given pixel \(p_i\) can be expressed as a seven-dimensional vector \(\mathbf{V}(p_i)\).

\[
\mathbf{V}(p_i) = (\cos(h), \sin(h), s, v, x, y, T) \quad (15)
\]

where \((x, y)\) is the spatial coordinate of pixel \(p_i\), and \(T\) is the texture feature obtained by Equation (14).

Kernel Function Calculation: The calculation formula of the kernel function given in this method is as follows.

\[
\kappa(i, j) = 1 - \frac{X(i) - X(j)}{7} \quad (16)
\]

So we can further get the similarity matrix \(A\).

\[
A = [\kappa(i, j)]_{N \times N} \quad (17)
\]

Closed Form Solution: When the user enters the trimap, the closed-form is extracted to solve the equation, as follows.

\[
(L + \lambda M) \alpha = \lambda m \quad (18)
\]

where \(L\) is the Laplace matrix \(L = D - A\) \((D = \text{diag}(D))\), \(M = \text{diag}(m)\), \(m\) is the binary vector of all marked pixels, and the constant \(\lambda\) is the constraint coefficient. The closed solution for the optimisation function is shown in Equation (19).

\[
g(x) = x^T N x + \lambda \sum_{i \in m-v} x_i^2 + \lambda \sum_{i \in v} (x_i - 1)^2 \quad (19)
\]

\[
\frac{\delta g}{\delta x} = H x - c = 0 \quad (20)
\]

The final solution is shown in Equation (21).

\[
x^* = H^{-1} c = (L + \lambda M)^{-1} (\lambda v) \quad (21)
\]

### 3.2 Image attention retargeting algorithm

Generally, colours are described in terms of hue, saturation and brightness. Hue describes the pure colour attribute, saturation describes the measure of the degree to which a pure colour is diluted by white light, and brightness is the key factor to describe the colour and is measurable. When colour is mentioned, the first thing generally thought of is the hue in the colour, while the saturation and brightness are not as noticeable. However, in image saliency detection, the saturation and brightness of the object colour have a greater impact on the saliency. Therefore, to get a better effect of modifying the significance, HSI colour space is used to modify the ROI area from three aspects, hue, saturation and brightness, and the brightness component is separated from the colour information in the image to achieve the best effect of reducing saliency.

Suppose a natural image is obtained with hue, saturation, brightness, and its selected ROI area. Since the lowest level of
saliency is determined by the local saliency in the main visual features, the following conclusion can be drawn: To minimise the saliency of the ROI area, the ROI area is almost identical to the surrounding background area. First, without considering the saturation S and brightness I, the hue of the ROI is adjusted to the background. Of course, natural images are not composed of one tone. A convenient representation is the hue distribution histogram. In the hue distribution, more than 50% of the hues can be taken as the background hue. The task is to obtain the hue distribution that is most similar to the surrounding background.

The hue representation is uncertain, so the modelled hue information needs to consider the function \( u = q(h_1, h_2, \ldots, h_n) \) of the measured value. For an independent measurement with a corresponding standard deviation \( \sigma_u, \ldots, \sigma_u \), the uncertainty in \( u \) is given by \([44]\).

\[
\sigma_u = \sqrt{\sum_{i=1}^{n} \left( \frac{\partial q}{\partial h_i} \sigma_{u_i} \right)^2}
\]

(22)

Applying it to hue \( b \), the relative metric \( \sigma_b = \frac{\rho}{\sigma} \) about the uncertainty of each image is obtained, where \( \rho \) is the maximum hue of all pixels in the image, and the range of \( \sigma_b \) will be limited to \([1, \infty]\).

As for hue \( b \in [0, 2\pi) \), kernel density estimation is used to construct the hue distribution, as shown in Equation (23).

\[
P(b) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{\sigma_b} \kappa\left(\frac{b - b_i}{\sigma_b}\right) p_i
\]

(23)

where \( N \) is the total number of pixels in the range, \( b_i \) is the hue at pixel \( i \), and \( p_i \) is the prior probability of \( b_i \). In the choice of kernel function, Dirac kernel function is preferred as it can convert the kernel density estimation in the above formula into a simple weighted histogram with a fast calculation speed. In the ROI region, it is believed that each hue observation value has equal probability, which is \( p_i = 1/N \). For the background area, it was modelled by convolving the Gaussian function, and \( \sum_{i=1}^{N} p_i = 1 \) is made according to normalisation.

From the above process, the hue distribution of the ROI area and the surrounding background area as \( P_R(b) \) and \( P_K(b) \), respectively, are obtained. According to the above requirements, it is necessary to make the ROI area and the surrounding background area have a similar hue distribution to the greatest extent possible, that is, \( P_R(b) \) and \( P_K(b) \) must be as similar as possible. Therefore, the difference between them needs to be measured and expressed with relative entropy \( G(\theta) \),

\[
G(\theta) = \int_{0}^{2\pi} P_R(b) \log \left( \frac{P_R(b)}{P_K(b + \theta)} \right) db.
\]

(24)

To obtain the most similar hue distribution, the optimal case is to minimise \( G(\theta) \), that is \( \theta_{adj} = \arg \min G(\theta) \). Dirac kernel function is used, which may occur that with multiple different \( \theta \) values meeting the requirements in a flat peak segment, so the mean value of the range is used as the optimal solution \( \theta_{adj} \). Therefore, all hues in the ROI area are adjusted to \( b_i = b_i + \theta_{adj} \), and the value range of hue \( b \) is \([0, 2\pi]\), so when \( \hat{b}_i \geq 2\pi \), \( \hat{b}_i = \hat{b}_i - 2\pi \).

For saturation and brightness channels, they were further adjusted, through the interaction parameters saturation difference \( \Delta S \) and brightness difference \( \Delta I \), which represent the saturation difference and the brightness difference, respectively, between the ROI area and the background area. Because these two parameters are directly related to the image saliency, their effect is obvious, and they are easy to adjust.

For input natural image \( M \), to express the influence more clearly by mathematical formula, about the user control parameters \( \Delta S \) and \( \Delta I \) on the image, two functions are defined, \( \Phi(S_M, R) \) and \( \Phi(I_M, R) \), and used to calculate the saturation difference and brightness difference between the ROI area and the background area:

\[
\Phi(S_M, R) = \text{mean } \{ S_M \notin R \} \text{ mean } \{ S_M \in R \},
\]

(25)

\[
\Phi(I_M, R) = \text{mean } \{ I_M \notin R \} \text{ mean } \{ I_M \in R \}.
\]

(26)

where \( M \) is the processed image, and initially \( M = M \). The corresponding saturation and brightness are represented by \( S_M \) and \( I_M \), and \( R \) is the ROI area.

The energy terms are set based on saturation and brightness, and the difference between saturation and brightness is transformed into the minimum energy problem,

\[
E(s) = \| \Phi(S_M, R) - \Delta S \|,
\]

(27)

\[
E(i) = \| \Phi(I_M, R) - \Delta I \|.
\]

(28)

To improve the robustness to outliers, in the calculation of the average value, only the saturation and brightness that \( \alpha_{top} (=20\% \text{ of the ROI area and the background are considered. When } E(s) = \| \Phi(S_M, R) - \Delta S \| < \varepsilon_s \text{ and } E(i) = \| \Phi(I_M, R) - \Delta I \| < \varepsilon_i \text{ are satisfied, the energy function converges, that is, the set difference is reached. If the convergence fails, the iteration stops when the energy item no longer changes during the subsequent iterations, and the optimal solution is obtained. In the experiment, the default value of the two differences is defined as } \Delta S = \Delta I = 0.6. \text{ Since the range of threshold is not clearly limited, they are normalised and the default value set to } \varepsilon_s = \varepsilon_i = 0.05. \]

4 | EXPERIMENTAL EVALUATION

In this section, the setup and results of the experiment to assess the performance of this method are discussed.
4.1 | Data set selection

In the selection of experimental data sets, at least two requirements must be met: (1) each image has at least one saliency object; (2) each group contains at least one image, in which there are two or more saliency targets, and each object covers at least 10 percent of the image. This can prevent the most significant problems caused by excessive ROI area or the neglected problem caused by the ROI area being too small.

According to the above requirements for experimental data, 20 groups of images were selected from CMU, Cosal2015 and iCoseg, totalling 100 images for experimental verification.

4.2 | Image preprocessing

The following is the construction phase of the saliency-guided image block (i.e. ROI). First, a cluster-based co-saliency detection algorithm was used to estimate the co-saliency map of all image groups and the single saliency map of each image. By comparing the co-saliency map and single saliency map, the rough position of the ROI area can be estimated. Then, through the matting algorithm, the grey image of non-common saliency objects can accurately and completely be obtained, that is, the saliency-guided image block. Here, to increase the accuracy of ROI region extraction, the method of manually drawing the trimap was used. Figure A1 in the Appendix shows the partial results of the saliency detection. As shown in Figure 4, part of the extraction results of matting are shown. Figure 5 is an enlarged detail of the ROI region. It can be seen that these methods can accurately judge the foreground and background, and even can be accurately detected on the small tip of the tomato pedicle. However, there remains room for improvement in the handling of edge details.

4.3 | Evaluation standard

A qualitative and quantitative analysis has been performed. Qualitative analysis uses the intuitive perception of human eyes to evaluate the effect of visual attention retargeting processing. By comparing the saliency maps of the single image before and after the processing with the co-saliency maps, respectively, quantitative analysis to obtain specific evaluation index values is more convincing.

4.3.1 | Qualitative analysis

In terms of image attention retargeting, a questionnaire survey was made for 20 groups of the original images and the images processed using the described method. It contains three questions: (1) In the original image, which is the most significant object? (2) In the processed image, which is the most significant object? (3) Are the most prominent objects in the two images the same? In the first two questions, several candidate objects are supplied for testers to choose from. A total of 53 testers participated in the survey. The most significant objects in the two images selected by 71.6% of the testers are different. In the second question, 56.6% of the testers did not select the ROI area as the salient object. Experimental results show that this image processing changed the saliency of the image to a certain extent, and accurately matches the ROI area that was extracted.

4.3.2 | Quantitative analysis

To more intuitively embody the saliency reduction effect of this method on non-common salient objects, the original image co-saliency map was used as the reference truth map. The original single-image saliency map and the single-image saliency map modified using the described method are used as test maps. Saliency evaluation index F-Measure, Mean Absolute Error (MAE), P–R curve, Matrix Similarity (Similarity, SIM), Normalised Scanpath Saliency (NSS), Linear Correlation
Coefficient (CC), and KL divergence (KLdiv) were selected to evaluate the experimental effect [45]. When calculating F-Measure, $\beta^2 = 0.3$ was taken, and the results of various indicators are shown in Table 1 and Figure 6. Figure 7 is the P–R curve of the original single-image saliency map and the modified single-image saliency map.

From the evaluation indicators in Figure 6, Figure 7, and Table 1, the visual attention retargeting algorithm based on co-saliency detection our proposed, in each index, reflects the modified single-image saliency map is closer to the original image co-saliency maps, and this method effectively reduces the saliency of non-common saliency objects.

In the comparative experiment, the results from this method were compared with those of Su et al. [35], SaGIM [42], and the OHA algorithm [46], and the partial experimental results are shown in Figure 8. More experimental results are shown in Figure A2 in the Appendix. Among them, the first column of each group is the processing result of the method, and the second column is the single-image saliency map of the result. The described method has obvious advantages in reducing saliency and effectively suppresses the saliency characteristics of the ROI region. Besides, for the attention retargeting effect, the colour of the image obtained using the described method is more natural, and is adjusted on the entire ROI, while the other algorithms can only modify some areas in the ROI area in individual images. As shown in the kites in the example on the fourth row in Figure 8, the white area in the original image cannot be modified. Moreover, the methods in [35,42] seriously affect the quality of the image, resulting in a decrease in the definition of the ROI area in the image, while the method described here shows good performance in image quality maintenance. In the experiment, the authors found that the results of this method on the image of people are not perfect. For example, there may be unrealistic colours, such as green players, as shown in the 16th row of Figure A2. In the next study, this problem needs to be resolved.

5 | CONCLUSIONS AND FUTURE WORK

The authors propose an image attention retargeting algorithm based on co-saliency detection. The co-saliency detection algorithm is used to obtain the approximate position of non-common salient objects, and then the matting algorithm is used to accurately and completely extract their precise location. Finally, the authors propose a new method of attention redirection. In the HSI colour space, they describe the hue content of an ROI and its surroundings using a polar representation of a perceptually uniform colour space, so that the best hue adjustment to minimise the difference between the ROI and its surrounding environment can be easily determined. In terms of saturation and brightness, the parameters can be set according to the user’s ROI area and the surrounding environment differences, which is more user-friendly and the effect is much better. In future work, the authors plan to optimise the
accuracy of the matting algorithm in image preprocessing to make it more accurate in small parts. The attention retargeting algorithm also needs to continue to be optimised for the naturalness of colour, while reducing the ROI saliency, making the ROI region features more realistic.

ACKNOWLEDGEMENTS
This research was supported by the National Natural Science Foundation of China (61772319, 62002200, 61976125, 61976124, 61907026), Project of Shandong Province Higher Educational Science and Technology Program (J18KA392) and Project of Shandong Technology and Business University wealth management (2019ZBKY053, 2019ZBKY032).

ORCID
Jinjiang Li https://orcid.org/0000-0002-2080-8678

REFERENCES
1. Fu, H.Z., et al.: Object-based multiple foreground video Co-segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Columbus, 3166–3173 (2014)
2. Hadizadeh, H., Bajic, J.V.: Saliency-aware video compression. IEEE Trans Image Process. 23(1), 19–33 (2014)
3. Fu, H.Z., Cao, X.C., Tu, Z.W.: Cluster-based Co-saliency detection. IEEE Trans. Image Process. 22(10), 3766–3778 (2013)
4. Nie, D.D., Wang, L.: Improved KNN matting method. J. Chin. Comput. Syst. 36(06), 1316–1320 (2015)
5. Chen, H.T.: Preattentive co-saliency detection. In: IEEE international conference on image processing, 1117–1120. Hong Kong: IEEE (2010)
6. Cao, X.C., et al.: Self-adaptively weighted co-saliency detection via rank constraint. IEEE Trans Image Process. 23(9), 4175–4186 (2014)
7. Han, J.W., et al.: A unified metric learning-based framework for cosaliency detection. IEEE Trans. Circ. Syst. Video Technol. 28(10), 2473–2483 (2018)
8. Li, L.N., et al.: Co-saliency detection based on region-level fusion and pixel-level refinement. In: IEEE International Conference on Multimedia and Expo. Chengdu, 1–6 (2014)
9. Tsai, C., et al.: Image co-saliency detection and co-segmentation via progressive joint optimization. IEEE Trans Image Process. 28(1), 56–71 (2019)
10. Zhao, J., et al.: FLIC: fast linear iterative clustering with active search. Comput. Vis. Media. 4(4), 49–64 (2018)
11. Tan, Z., et al.: Image co-saliency detection by propagating superpixel affinities. In: IEEE International Conference on Acoustics, Speech and signal processing, 2114–2118. Vancouver (2013)
12. Liu, Z., et al.: Co-saliency detection based on hierarchical segmentation. IEEE Signal Process. Lett. 21(1), 88–92 (2014)
13. Li, Z., et al.: Co-saliency detection with graph matching. ACM Trans. Intell. Syst. Technol. 10(3), 1–22 (2019)
14. Zhang, K., et al.: Co-saliency detection via mask-guided fully convolutional networks with multi-scale label smoothing. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 3095–3104. Long Beach (2019)
15. Zhang, D., et al.: Co-saliency detection based on intra-saliency prior transfer and deep intersaliency mining. IEEE Trans. Neural Netw. Learn. Syst. 27(6), 1163–1176 (2016)
16. Zhang, D., et al.: Co-saliency detection via looking deep and wide. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2994–3002. IEEE, Boston (2015)
17. Zhang, D., et al.: A self-paced multiple-instance learning framework for co-saliency detection. In: Proceedings of the IEEE International Conference on Computer Vision, 594–602 (2015)
18. Tsai, C.C., et al.: Deep co-saliency detection via stacked autoencoder-enabled fusion and self-trained cnns’. IEEE Trans. Multimedia. 22(4), 1016–1031 (2019)
19. Gao, G., et al.: Co-saliency detection with Co-attention fully convolutional network. IEEE Trans Circuits Syst Video Technol. (2020)
20. Bi, H., et al.: C²Net: a complementary co-saliency detection network. Vis. Comput. (2020). https://doi.org/10.1007/s00371-020-01842-4
21. Zha, Z.J., et al.: Robust deep Co-saliency detection with group semantic and pyramid attention’. IEEE Trans Neural Netw Learn Syst. 31(7), 2398–2408 (2020)
22. Porter, T., Duff, T: Compositing digital images. ACM Siggraph Comput. Graph. 18(3), 253–259 (1984)
23. Ruzon, M.A., Tomasi, C.: Alpha estimation in natural images. In: Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 18–25. Hilton Head Island (2000)
24. Huang, H., et al.: Pixel-level discrete multiobjective sampling for image matting’. IEEE Trans Image Process. 28(8), 3739–3751 (2019)
25. Liang, Y., et al.: Multiobjective evolutionary optimization based on fuzzy multicriteria evaluation and decomposition for image matting. IEEE Trans. Fuzzy Syst. 27(5), 1100–1111 (2019)
26. Sun, J., et al.: Poisson matting. ACM Trans. Graph. 23(3), 315–321 (2004)
27. Wang, J., Cohen, M.F.: Optimized color sampling for robust matting. In: IEEE Conference on Computer Vision and Pattern Recognition, 1–8. Minneapolis (2007)
28. Chen, Q., Li, D., Tang, C.K.: KNN matting. IEEE Trans. Pattern Anal. Mach. Intell. 35(9), 2178–2188 (2013)
29. Aksoy, Y., Aydin, T.O., Pollefeys, M.: Designing effective inter-pixel information flow for natural image matting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, 228–236 (2017)
30. Cho, D., Tai, Y.W., Ko, I.: Natural image matting using deep convolutional neural networks. In: European Conference on Computer Vision, 626–643. Cham. Springer (2016)
31. Xu, N., et al.: Deep image matting. In: Proceedings of the IEEE Conference on Computer Vision And Pattern Recognition, 2970–2979(2017)
32. Lutz, S., Amplaniitis, K., Smolic, A.: AlphaGan: generative adversarial networks for natural image matting. arXiv preprint (2018)
33. Lu, H., et al.: Indices matter: learning to index for deep image matting. In: Proceedings of the IEEE International Conference on Computer Vision. Seoul, 3266–3275 (2019)
34. Tang, J., et al.: Learning-based sampling for natural image matting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3055–3063. Seoul (2019)
35. Su, S.L., Durand, F., Agrawala, M.: De-emphasis of distracting image regions using texture power maps. In: Texture 2005: Proceedings of the 4th ICCV Workshop on Texture Analysis and Synthesis, 119–124. Beijing (2005)
36. Hagiwara, A., Sugimoto, A., Kawamoto, K.: Saliency-based image editing for guiding visual attention. In: Proceedings of the 1st International Workshop on Pervasive Eye Tracking & Mobile Eye-Based Interaction, 43–48. New York (2011)
37. Li, J., Feng, X., Fan, H.: Saliency-based image correction for colorblind patients. Comput. Vis. Media. 6(2), 169–189 (2020)
38. Li, J., Feng, X., Fan, H.: Saliency consistency-based image re-colorization for color blindness. IEEE Access. 8, 88558–88574 (2020)
39. Mechrez, R., Shechtman, E., Zelnik-Manor, L.: Saliency driven image manipulation. Mach. Vis. Appl. 30(2), 189–202 (2019)
40. Hata, H., Koike, H., Yoichi, S.: Visual attention guidance using image resolution control. In: Proceedings of IPSJ Interaction, 57–64 (2014)
41. Hitomi, S., et al.: Guiding visual attention using saliency map based on wavelet transform. In: Proceedings of RISP International Workshop on NCS, 90–93 (2015)
42. Chen, Y.C., et al.: Guide your eyes: learning image manipulation under saliency guidance Supplementary materials. In: British Machine Vision Conference (BMVC). Cardiff (2019)
43. Lee, P., Wu, Y.: Nonlocal matting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Providence, 2193–2200 (2011)
44. Taylor, J.R., Thompson, W.: An introduction to error analysis: the study of uncertainties in physical measurements. Am. J. Phys. 51(1), 57–58 (1997)
45. Bylinskii, Z., et al.: What do different evaluation metrics tell us about saliency models? IEEE Trans. Pattern. Anal. Mach. Intell. 41(3), 740–757 (2019)
46. Mateescu, V.A., Bajić, I.V.: Attention retargeting by color manipulation in images. In: Proceedings of the 1st International Workshop on Perception Inspired Video Processing, 15–20. New York: Association for Computing Machinery (2014)

How to cite this article: Ren X, Li J, Hua Z, Jiang X. Consistent image processing based on co-saliency. CAAI Trans. Intell. Technol. 2021;1-14. https://doi.org/10.1049/cit2.12020
**FIGURE A1** Example of cluster-based saliency detection results
Figure A2  More examples of image attention retargeting results by the authors' method and single-image saliency map, and comparative experiments include OHA [46], SaGIM [42], Su et al. [35]