Multimedia Tools and Applications (2020) 79:14419–14447
https://doi.org/10.1007/s11042-018-6770-2

Learning adaptive contrast combinations for visual saliency detection

Quan Zhou[1,2] · Jie Cheng[3] · Huimin Lu[4] · Yawen Fan[1] · Suofei Zhang[5] · Xiaofu Wu[1] · Baoyu Zheng[1] · Weihua Ou[6] · Longin Jan Latecki[7]

Received: 2 August 2018 / Revised: 25 September 2018 / Accepted: 10 October 2018 /
Published online: 7 November 2018

Abstract

Visual saliency detection plays a significant role in the fields of computer vision. In this paper, we introduce a novel saliency detection method based on weighted linear multiple kernel learning (WLMKL) framework, which is able to adaptively combine different contrast measurements in a supervised manner. As most influential factor is *contrast* operation in bottom-up visual saliency, an *average weighted corner-surround contrast* (AWCSC) is first designed to measure local visual saliency. Combined with common-used *center-surrounding contrast* (CESC) and *global contrast* (GC), three types of contrast operations are fed into our WLMKL framework to produce the final saliency map. We show that the assigned weights for each contrast feature maps are always normalized in our WLMKL formulation. In addition, the proposed approach benefits from the advantages of the contribution of each individual contrast feature maps, yielding more robust and accurate saliency maps. We evaluated our method for two main visual saliency detection tasks: human fixed eye prediction and salient object detection. The extensive experimental results show the effectiveness of the proposed model, and demonstrate the integration is superior than individual subcomponent.

Keywords Saliency detection · Contrast combinations · Visual attention · Multiple kernel learning

1 Introduction

The human visual system (HVS) has an outstanding ability to quickly locate the most interesting parts in a given scene. Such image parts are considered as salient since it is assumed these parts attract greater attention than other parts by the HVS. The recent study of saliency approaches may reveal the attention mechanisms of visual biology to predict human fixa-

✉ Quan Zhou
quan.zhou@njupt.edu.cn
✉ Suofei Zhang
zhangsuofei@njupt.edu.cn

Extended author information available on the last page of the article.

tion selection behavior, as well as finding most salient objects/regions that are outstanding from the backgrounds. Saliency detection is involved in many visual applications, such as automatic image cropping [10], image thumbnailing [44], image/video compression [19], video popularity prediction [28], image memorability estimation [29], image segmentation [14], image quality assessment [42], and object detection/recognition [2, 16]. The recent years have witnessed great progress in visual saliency detection, and it has received extensive attention by the researcher in the fields of psychologists and computer vision [1, 7, 10, 17, 20, 21, 24, 25, 31, 49, 55, 58, 67]. As a pioneer work, Treisman [60] proposed a feature integration theory (FIT) which is composed by three main steps for HVS: (1) the bottom-up contrast computation based on simple low-level image stimuli signals, such as luminance, color, texture and orientation, which are driven from the input image [1, 25, 41]; (2) the integration process via fusing various bottom-up feature maps produced in first step [18]; (3) the enhanced highlighting salient parts with the assistance of top-down priors if available [24, 31].

In spite of achieving promising results, these approaches are still suffered from the following limitations: (1) The traditional bottom-up approaches are mainly computed from center-surrounding contrast (CESC) [25, 31] to estimate local visual saliency. This hypothesis often fails when the contrast can not provide enough discrimination between center and surrounding regions, which always yields blurry saliency map and sometimes highlights backgrounds instead of salient parts [7, 25, 68]. (2) The existing methods for feature map integration, such as average operation [25], selective fusing operation [18], max or min operation [71], are not flexible enough and adaptive sufficiently. They are not able to assign adaptive weights to predict visual saliency, which reflect the confidence level of each individual feature map. In summary, there are two issues of primary importance to be considered in visual saliency formulation:

- How to formulate robust local *contrast* computation to accurately estimate saliency localization?
- How to integrate different kind of feature maps to highlight entire salient part in a given image?

This paper presents a novel saliency detection scheme based on weighted linear multiple kernel learning (WLMKL) framework to approach both questions. More specifically, our method firstly utilizes *average weighted corner-surround contrast* (AWCSC) to measure the saliency for each pixel. Except computing the appearance contrast, this contrast operation also considers the relative location between center and surrounding regions, which enables us to predict more exact location of the salient parts. Thereafter, two types of contrast measurement, CESC [71] and global contrast (GC) [10], are calculated as complementary feature maps. Finally, to further investigate the contribution of each feature map, a multi-cues integration framework is designed using our WLMKL scheme to predict visual saliency. The proposed WLMKL utilizes a weighted ℓ_2-norm linear support vector machine (SVM) to formulate gaussian kernels of contrast feature maps. Under WLMKL scheme, we formulate visual saliency as a binary classification task, and the weights for each feature map can be learned adaptively in a supervised manner. To optimize our WLMKL model, we design an EM-like procedure to alternatively update the model parameters and combined feature weights, where a closed-form solution can be obtained for updating feature weights. Although the ideas of combining different feature maps have been investigated in [25, 71] for the task of human eye fixation prediction, to our best knowledge, there are limited work that have been tested with salient object detection. The main contributions of this paper are mainly summarized as follows:

- We show the advantage of using AWCSC to measure local contrast. By establishing feature map under this criteria, we are able to encode the appearance contrast as well as relative location between center and surrounding region. We conducted AWCSC in several datasets and show how this contrast measurement lead to better location prediction for salient parts.
- We propose to use WLMKL paradigm to formulate visual saliency, motivated by assigning adaptive weights to integrate different feature maps. Due to the duality, an efficient algorithm is designed to solve our WLMKL problem with ℓ_2-norm regularization. The proposed model benefits from the advantages of each individual contrast feature map, while keeping the assigned weights are always normalized.
- We evaluate our approach for two main tasks of visual attention: human eye fixation prediction and salient object detection. Compared with previous models, the extensive experiences show that our method achieves better results both in terms of detection accuracy and implemental efficiency.

This paper is organized as follows: After a brief discussion of related work in Section 2, we elaborate the detail of the proposed visual saliency detection method in Section 3. Section 4 reports the experimental results, and the conclusive remarks are given in Section 5.

2 Related work

Since the relevant literature is quite extensive, we review the related work instead emphasizes the key concepts crucial to the establishment of the proposed framework.

2.1 Bottom-up visual saliency detection

The contrast computation from low-level image stimuli is a significant factor in bottom-up saliency estimation. As a result, considerable efforts have been devoted to measure contrast in this category [21, 25]. For example, in [25], a center-surround contrast is employed to encode local saliency within different scales, which is inspired by the putative neural mechanism. It often intrinsically hypothesizes that the visual inputs are salient in certain background context. Similarly, saliency is defined as the local complexity [32] and self-information is formulated to predict local saliency in [7]. Gao et al. employed the Kullback-Leibler divergence (KLD) to measure the visual difference between center and its surroundings [15]. Ma et al. adopted a fuzzy growing algorithm to detect salient regions [41].

An alternative approach for highlighting salient part is to use global contrast with respect to the entire image [1, 10, 20, 49]. For instance, Cheng et al. utilized the contrast from local pixel/region against with other pixels/regions to formualte saliency [10]. Herel et al. [20] estimated saliency based on a graph formulation where the local saliency is computed depend on global information. In [1], the authors computed saliency based on the difference between the *Lab* pixel value and average *Lab* value of the entire image. Perazzi et al. [49] employed color uniqueness regarding to the whole image to calculate saliency. The global contrast can be also formulated in a spectral domain [21]. Unlike these models, we propose to use AWCSC to compute local contrast. In addition to address appearance contrast as well as CESC approaches does, AWCSC also considers the relative location between center and surrounding regions, yielding more accurate prediction for salient regions with less noises from the backgrounds.

2.2 Top-down visual saliency detection

Some successful systems has been constructed to capture top-down information to detect visual saliency [5, 51, 59]. As a pioneer work, Torralba et al. [59] employed the scene global information in their model construction. In [5], a series of discriminative classifiers (e.g., regression, SVM, and Adaboost) are trained to combine low-level features and top-down cognitive visual cues (e.g., faces, cars, humans, etc.) to predict human eye fixations. On the other hand, the top-down semantic features can be also used for salient object detection. Based on the observation that humans tend to look at the center of images [46], Chang et al. [9] proposed an object-based saliency model using an objectness measurement [2].

Recently, benefitting from the great success of powerful image feature representation, deep neural networks (DNNs) have been introduced in visual attention for salient object detection [36, 63, 64, 69, 70] and eye fixation prediction [23, 34, 37]. In [11, 23, 27, 34, 37, 48, 69], the fully convolutional networks (i.e., VGGnet), which transfer fully connect layer into convolutional layer, has been utilized to estimate saliency location in each image. In [69], the multiple scale convolutional features in VGG-16 network are explored to detect salient object detection. To open "black-box" learning of previous deep neural networks, Zhang et al. [70] designed an deep uncertain convolutional features for both salient object detection and human eye fixation task, where an uncertain ensemble of internal feature units is construct in deconvolutional network. In [63], the salient regions are abstracted in a weakly supervised scenario that only image level tags are used as supervised information. However, all these previous works assessed saliency in local contexts due to their local features and pixel-wise classifiers [35, 37] or limited receptive fields in FCN based models [11, 23, 27, 34, 37, 48].

Although these approaches achieve promising results for human eye fixation prediction and salient object detection, respectively, to our best knowledge, there are little work to approach these two visual attention task simultaneously. Conversely, our method is adapted to both visual attention tasks and investigates the role of each individual map for final saliency producing, where the corresponding feature weights are always normalized.

2.3 Multiple kernel learning (MKL)

MKL refers to a kernel learning machine that combines inputs from various image features [8, 57]. Recent research efforts on MKL have shown that using a specific kernel feature may be a source of bias, a better solution may be found in allowing a learner to choose in an ensemble kernel feature functions [45]. Therefore, instead of trying to find which kernel works best, MKL attempts to find appropriate feature combinations among a set of candidate kernels [65]. Unlike previous MKL model, our WLMKL employs a weighted ℓ_2-norm linear SVM to formulate visual saliency, yielding a linear combination of an ensemble of gaussian kernel function based on contrast feature maps. Iteratively optimizing over the model coefficients and kernel feature weights is one particular form in our WLMKL model. Such optimization criteria enables us to find an optimal way to linearly combine the given contrast feature maps, yielding more accurate saliency detection results.

In this paper, we extend our first published work [50] in three aspects: we consider the relative location in corner-surrounding contrast calculation, while the previous version only considers the appearance contrast; we employ an WLMKL scheme for visual saliency detection, which can learn adaptive weights for different feature maps; we have implemented more complete experiments for human eye fixation prediction as well as salient object detection, and reported more comparisons and higher results.

3 The proposed method

The diagram of our approach is shown in Fig. 1. The training and testing images undergo the same procedure of feature map computation and combination. More specifically, three contrast operations are applied to each sub-channel of RGB color format separately. Following [25, 26, 31], the rarity of image patches is first employed to compute CESC with respect to their surrounding neighborhoods. Besides the difference of appearance, the second contrast operation, AWCSC, also considers the difference of location between center patch and its surroundings. The final operation, called GC, calculates the rarity of a feature or a region with respect to the entire image. Based on these contrast feature maps, we construct a WLMKL model to learn the weights for each individual feature map. Then, the proposed fusion method will combine the feature maps of testing images to produce the final saliency with the guidance of leaned feature weights. In this section, we first introduce image representation that is used to predict visual saliency on a patch-by-patch basis.

3.1 Image representation

From the perspective of human vision, a vision system should be adapted to the visual environment. As a supporting evidence for this theory, it has been shown that some neurons in V1 cortex resemble receptive fields that are learned via sparse coding algorithms [47]. From the perspective of computer vision, natural images are always with redundant structure, and

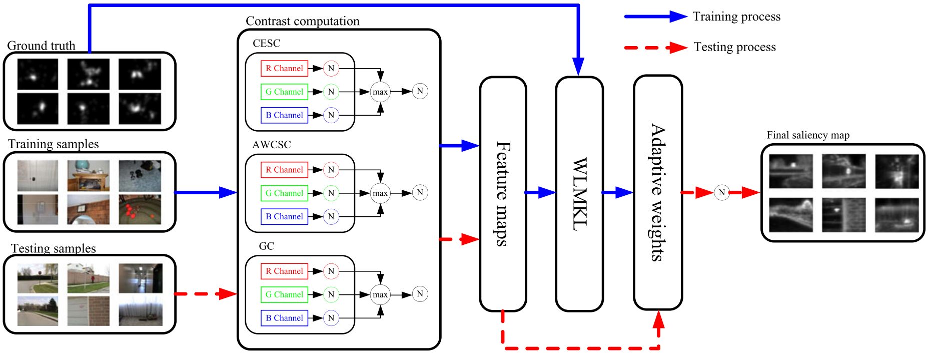


Fig. 1 Diagram of our saliency detection approach. In each channel of RGB color space, the AWCSC and CESC feature maps, which are the dissimilarity between a patch and its surrounding window, and global feature map, which is based on rarity of an image patch with respect to the entire scene, are computed and normalized for training images. Then, the output feature maps and corresponding ground truth are used to train our WLMKL model. Finally, the feature maps of testing image are fed into the learned WLMKL model to generate final saliency map. The blue arrow denotes the training process and the dash red arrow indicates the testing process. (Best viewed in color)

thus can be sparsely represented by a series of image basis functions [52]. To this end, we represent image patches using sparse coding technique, which has been widely used for saliency detection task [7, 22, 68]. Each image patch $\mathbf{p}_i$ is projected into the feature space spanned by a series of over-completed basis functions, which are learned from a repository of natural scenes. Then $\mathbf{p}_i$ can be encoded using the sparse constructed coefficients $\boldsymbol{\eta}_i$.

Preciously, the input image is first resized to $2^9 \times 2^9$ pixels. Let $\mathbf{P} = \{\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_N\}$ denote N image patches that has no overlap with each other. According to [47], the reconstructive coefficients $\boldsymbol{\eta}_i$ of patch $\mathbf{p}_i$ can be calculated as:

$$\boldsymbol{\eta}_i^*(\mathbf{p}_i, \mathbf{D}) = \arg \min_{\boldsymbol{\eta}_i \in \mathbb{R}^n} \frac{1}{2}||\mathbf{p}_i - \mathbf{D}\boldsymbol{\eta}_i||_2^2 + \lambda||\boldsymbol{\eta}_i||_1 \tag{1}$$

where λ is a regularization parameter and $|| \cdot ||_1$ denotes the ℓ_1-norm, which is used to constrain the reconstructive coefficients $\boldsymbol{\eta}_i$ are as sparse as possible. Let $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \cdots, \mathbf{d}_n] \in \mathbb{R}^{m \times n}$ be the visual dictionary, which is required to be learned from image datasets. It is composed by n basis functions, where each component $\mathbf{d}_i \in \mathbf{D}$ is a m-dimensional image patch. Thus, $\mathbf{p}_i \sim \mathbf{p}_i' = \mathbf{D}\boldsymbol{\eta}_i^*$, where $\mathbf{p}_i'$ is the constructed patch with smallest image residue of $\mathbf{p}_i$.

In order to learn the dictionary set $\mathbf{D}$, given q training samples $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_q] \in \mathbb{R}^{m \times q}$, an empirical cost function $g_q(\mathbf{D}) = \frac{1}{q}\sum_{i=1}^{q} l_u(\mathbf{y}_i, \mathbf{D})$ is minimized, where $l_u(\mathbf{y}_i, \mathbf{D})$ is defined as:

$$l_u(\mathbf{y}_i, \mathbf{D}) = \min_{\boldsymbol{\eta} \in \mathbb{R}^n} \frac{1}{2}||\mathbf{y}_i - \mathbf{D}\boldsymbol{\eta}_i||_2^2 + \lambda||\boldsymbol{\eta}_i||_1 \tag{2}$$

In our implementation, 500,000 image patches are randomly selected from color images, and each one has 8×8 resolution. We learn $n = 200$ basis functions, and thus each basis function $\mathbf{d}_i$ in the dictionary $\mathbf{D}$ is a $8 \times 8 = 64D$ vector. Given the learned visual dictionary $\mathbf{D}$, the sparse coding coefficients $\boldsymbol{\eta}_i$ are computed using the LARS algorithm [43].

3.2 Computing contrast feature maps

In this section, we first elaborate on the details of three contrast operations in Fig. 1, and then introduce how to normalize contrast feature maps.

3.2.1 CESC

Let $\mathbf{p}_i$ be the center patch. Inspired from the well-established computational architecture of [25, 33], the CESC operation, denoted as $f_{ce}^c(\mathbf{p}_i)$ in our model, is defined as the average weighted dissimilarity between $\mathbf{p}_i$ and its L surroundings:

$$f_{ce}^c(\mathbf{p}_i) = \frac{1}{L}\sum_{m=1}^{M} \mathbf{W}_{im}^{-1}\mathbf{B}_{im} \tag{3}$$

where $\mathbf{B}_{im}$ measures the appearance dissimilarity between $\mathbf{p}_i$ and its surround patch $\mathbf{p}_m$ using the Euclidean distance between $\boldsymbol{\eta}_i$ and $\boldsymbol{\eta}_m$, derived from sparse coding algorithm.

On the other hand, intuitively, those patches located far away from center will have less influence on saliency computation. Therefore, we adopt a location prior $\mathbf{W}_{im}$ to reflect this influence, where $\mathbf{W}_{im}$ is computed as the Euclidean distance between $\mathbf{p}_i$ and $\mathbf{p}_m$. Please note that the alternative distance measure, such as KLD [15], ℓ_1 norm [12], or correlation coefficient [72] can also be used to calculate patch similarity. Superscript c denotes sub channels in RGB color space.

3.2.2 AWCSC

As shown in Fig. 3, due to the fact that high saliency values are often assigned to background, using CESC might be not enough to detect saliency since it only explore very local dissimilarity with respect to certain background context. To overcome this drawback, AWCSC operation, denoted as $f_c^c(\mathbf{p}_i)$, is employed to further enhance visual saliency estimation. Compared with CESC, AWCSC not only explores the appearance dissimilarity between center patch $\mathbf{p}_i$ and the its surrounding neighborhoods, but also takes their relative location into account [50]. As shown in Fig. 2, four types of contrast templates, called top-right, top-left, bottom-right, and bottom-left, are defined. Let $f_{tr}^c(\mathbf{p}_i)$, $f_{tl}^c(\mathbf{p}_i)$, $f_{br}^c(\mathbf{p}_i)$, and $f_{bl}^c(\mathbf{p}_i)$ denote these four types of local contrast, respectively, the final AWCSC feature map is calculated as:

$$f_c^c(\mathbf{p}_i) = f_{br}^c(\mathbf{p}_i) \times f_{bl}^c(\mathbf{p}_i) \times f_{tr}^c(\mathbf{p}_i) \times f_{tl}^c(\mathbf{p}_i) \tag{4}$$

For one type of local contrast (e.g., $f_{br}^c(\mathbf{p}_i)$), we define it as well as CESC computation that encodes the discriminative difference between patch $\mathbf{p}_i$ and its surrounding region $\mathbf{s}_m$ (denote as red and blue cells, respectively, in the first subgraph of Fig. 2), synchronously considering their relative location:

$$f_{br}^c(\mathbf{p}_i) = \frac{1}{L}\sum_{m=1}^{L} \mathbf{W}_{im}^{-1}\mathbf{B}_{im} \tag{5}$$

The same operation then applies to $f_{bl}^c(\mathbf{p}_i)$, $f_{tr}^c(\mathbf{p}_i)$ and $f_{tl}^c(\mathbf{p}_i)$, separately.

From (4), it is evidence that AWCSC assigns high value to patch $\mathbf{p}_i$ only when it is recommended by four type of local contrast simultaneously. Thus AWCSC is a more strict contrast operation than CESC, resulting in more effective to exclude outliers and inhibit background, as shown in Fig. 3.

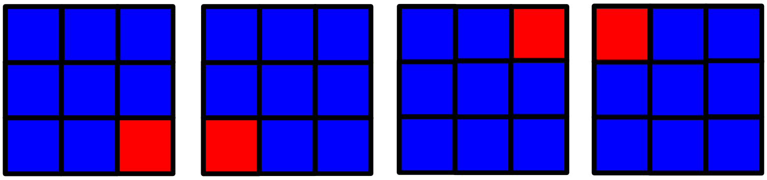


Fig. 2 Four types of AWCSC. Each cell is an image patch with resolution of 8 × 8, where the center patch and the corresponding surrounding patches are represented by red and blue cells, respectively. (Best viewed in color)

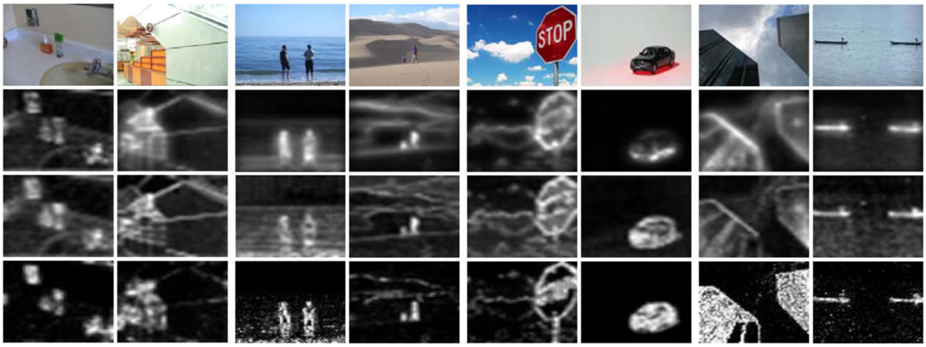


Fig. 3 Some visual examples of feature maps using different contrast computation. From left to right are samples from TORONTO, MIT, ASD and SED datasets, where each dataset has two sample images. The first two datasets are used to evaluate the human eye fixation prediction, while the later two datasets are employed to evaluate salient object detection. From up to bottom are original images and corresponding feature maps using CESC, GC, and AWCSC, respectively. (Best viewed in color)

3.2.3 GC

Sometimes, only using the local contrast operation may be unable to uniformly highlight the entire saliency part in an image. Especially for object-based attention, the salient parts within homogeneous regions are often suppressed, and object boundaries are often overemphasized. As shown in Fig. 3, CESC and AWCSC often lead to the blank holes in the detection results. Although local patch and its neighbors may have similar appearance cues, with respect to the whole image, the local patches are globally rareness. To this end, we construct our global contrast feature map $f_g^c(\mathbf{p}_i)$ guided from the information-theoretic measure [7]. Instead of computing visual saliency using local contrast, here we calculate the inverse of probability $p(\mathbf{p}_i)$ for each patch over the entire scene as the global feature map:

$$f_g^c(\mathbf{p}_i) = p(\mathbf{p}_i)^{-1} = p(\boldsymbol{\eta}_i)^{-1} \tag{6}$$

Note patch $\mathbf{p}_i$ is represented by vector $\boldsymbol{\eta}_i$ using sparse coding algorithm [47], which to some extent guarantees that the components of $\boldsymbol{\eta}_i$ are conditionally independent from each other. Therefore, the probability $p(\boldsymbol{\eta}_i)$ can be factorized into each subcomponents as follows:

$$\begin{aligned} &\log(f_g^c(\mathbf{p}_i)) = -\log p(\mathbf{p}_i) = -\log p(\boldsymbol{\eta}_i) = -\sum_{j=1}^{n}\log(p(\eta_{ij})) \\ &f_g^c(\mathbf{p}_i) \propto -\sum_{j=1}^{n}\log(p(\eta_{ij})) \end{aligned} \tag{7}$$

where η_{ij} is the j^{th} component of vector $\boldsymbol{\eta}_i$. To construct the probability density function ($p(\eta_{ij})$), we first calculate histogram distribution with B bins for each component η_{ij} among all image patches in the scene, then the distribution is normalized. In (7), the product will get a small value if patch $\mathbf{p}_i$ is rare in one component of $\boldsymbol{\eta}_i$, leading to high global contrast value $f_g^c(\mathbf{p}_i)$ for $\mathbf{p}_i$ with respect to the whole image.

3.2.4 Normalization and multi-scale inhibition

For each feature map defined as CESC, AWCSC and GC, the contrast feature maps $f(\mathbf{x})$ are constructed by assigning contrast values of patch $\mathbf{p}_i$ to its contained pixel $\mathbf{x}$, then $f(\mathbf{x})$ is normalized into a fixed range [0, 1] following [50]:

$$f(\mathbf{x}) = \frac{f(\mathbf{x}) - f_{min}}{f_{max} - f_{min}} \tag{8}$$

where f_{max} and f_{min} are the minimum and maximum values, respectively, among the entire feature map $f(\mathbf{x})$.

Since salient parts may tend to with different scales, it is required to perform saliency detection at several spatial scales. To make our approach multi-scale, we calculate the contrast feature maps from the images downsampled from the original image and then take the max operation after normalization.

3.3 WLMKL framework

According to FIT [60], the fusing stage is able to benefit from the advantage of each bottom-up feature map, thus leading to highlight salient part and suppress distractors in an input image. In order to effectively combine different feature maps, we design a WLMKL framework to learn model parameters and adaptive feature weights simultaneously, where the visual saliency is estimated as a binary classification task. From the perspective of WLMKL, the feature maps are explicitly encoded through a set of so-called basic kernel functions $\{k_m\}_{m=1}^{M}$, then a SVM objective is employed to optimal model parameters and kernel feature weights.

3.3.1 Saliency formulation

In the scenarios of saliency detection, we consider a training set Ω containing N samples, $\Omega = \{(f_n(\mathbf{x}), y_n \in \pm 1)\}_{n=1}^{N}$, where $f_n(\mathbf{x})$ represents the contrast score for pixel $\mathbf{x}$, and is characterized by aforementioned three kinds of feature maps. y_n is the target label for pattern$f_n(\mathbf{x})$, where $+1$ denotes the pixel $\mathbf{x}$ is salient, and -1 indicates not. For notation simplicity, we use f_n to represent $f_n(\mathbf{x})$. From the perspective of MKL, the saliency formulation of the learning problem for pixel $\mathbf{x}$ is with the form:

$$s(\mathbf{x}) = \sum_{m=1}^{M} \langle \boldsymbol{\omega}_m, \phi_m(f) \rangle + b \tag{9}$$

where m indexes kernels, M is the total number of kernels, $\boldsymbol{\omega}_m$ is the weight coefficients, b is the threshold, $\phi_m(\cdot)$ is the implicit mapping function for feature space m, and $\langle \cdot, \cdot \rangle$ denotes inner product. As shown in [3], the optimality conditions require these $\boldsymbol{\omega}_m$ to be written as:

$$\boldsymbol{\omega}_m = \beta_m \sum_{n=1}^{N} \alpha_n y_n \phi_m(f_n) \tag{10}$$

where α_n is model coefficient required to be learned from training samples, and β_m is the feature weight. Keeping in mind that different feature maps (CESC, AWCSC, and GC) may have different proportion of contribution for final saliency map, we thus constrain the feature weights in (10) are nonnegative and always normalized, which satisfy $\beta_m \geq 0, \sum_{m=1}^{M} \beta_m = 1$. By plugging $\boldsymbol{\omega}_m$ derived from duality conditions into (9), we obtain:

$$\begin{aligned} s(\mathbf{x}) &= \sum_{n=1}^{N} \alpha_n y_n \sum_{m=1}^{M} \beta_m \langle \phi_m(f_n), \phi_m(f) \rangle + b \\ \beta_m &\geq 0, \qquad \sum_{m=1}^{M} \beta_m = 1 \end{aligned} \tag{11}$$

If we define the basic kernel function as: $k_m(f_n, f) = \langle \phi_m(f_n), \phi_m(f) \rangle$, we get our final discriminative saliency model as:

$$\begin{aligned} s(\mathbf{x}) &= \sum_{n=1}^{N} \alpha_n y_n \sum_{m=1}^{M} \beta_m k_m(f_n, f) + b \\ \beta_m &\geq 0, \qquad \sum_{m=1}^{M} \beta_m = 1 \end{aligned} \tag{12}$$

Through the ensemble kernel functions which are symmetric and positive defined, we are able to explore the feature space more efficiently. Note our formulation considers that the basic kernel functions are convex combined, where model parameters $\{\alpha_n\}$ and $\{\beta_m\}$ are required to be learned synchronously.

3.3.2 WLMKL primal learning problem

Actually, (12) corresponds to a standard SVM formulation under MKL framework [45]. It tries to find a series of hyperplanes $\langle \boldsymbol{\omega}_m, \phi_m(f) \rangle + b$ that has large margin and small training error. Mathematically, this leads to the following convex optimal problem, which we refer to as our primal WLMKL problem:

$$\begin{aligned} \min_{b,\, \boldsymbol{\xi},\, \boldsymbol{\beta},\, \boldsymbol{\omega}} \quad & \frac{1}{2} \sum_{m} \frac{1}{\beta_m} ||\boldsymbol{\omega}_m||^2 + C \sum_{n} \xi_n \\ s.t. \quad & y_n \left[\sum_{m} \beta_m \langle \boldsymbol{\omega}_m, \phi_m(f_n) \rangle + b \right] \geq 1 - \xi_n, \forall n \\ & \xi_n \geq 0, \forall n \\ & \beta_m \geq 0, \qquad \sum_{m} \beta_m = 1 \end{aligned} \tag{13}$$

where $\boldsymbol{\omega} = \{\boldsymbol{\omega}_m\}_{m=1}^{M}$, $\boldsymbol{\beta} = \{\beta_m\}_{m=1}^{M}$, $\boldsymbol{\xi} = \{\xi_n\}_{n=1}^{N}$ are slack variables, and C is the trade-off parameter between training error and margin. It is clear that (13) is a primal learning problem involved in a weighted ℓ_2-norm regularization, where β_m controls the shape of the objective function. The bigger β_m is, the sharper of objective function should be. In particular, when β_m approaches zero, $||\boldsymbol{\omega}_m||^2$ are also required to be zero, yielding a finite value to the objective function.

Since this primal formulation is convex and differentiable, it provides a simple derivation of the dual problem [62]. By simply setting zero to the derivatives of the Lagrangian function for (13) with respect to the primal variables, we derive the associated dual problem as follows:

$$\begin{aligned}
\max_{\boldsymbol{\alpha}} \min_{\boldsymbol{\beta}} J(\boldsymbol{\alpha}, \boldsymbol{\beta}) &= -\frac{1}{2} \sum_{n,n'} \alpha_n \alpha_{n'} y_n y_{n'} \sum_m \beta_m k_m(f_n, f_{n'}) \\
&\quad + \sum_n \alpha_n \\
s.t. \quad & \sum_n \alpha_n y_n = 0 \qquad 0 \leq \alpha_n \leq C \quad \forall n \\
& \beta_m \geq 0, \qquad \sum_m \beta_m = 1
\end{aligned} \tag{14}$$

where $\boldsymbol{\alpha} = \{\alpha_n\}_{n=1}^N$. Optimizing over both the coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ is one particular form of the proposed WLMKL problems. Our approach utilizes such optimization to yield more flexible feature integration for visual saliency estimation, which is introduced in Section 3.3.4.

3.3.3 Basic kernel function construction

The choice of basic kernel function in kernel-based methods plays an important role to design an effective SVM classifier. Although we have established and normalized the feature maps using CESC, AWCSC and GC, these maps are always statistical fluctuations within different feature domains. To eliminate such variances in representation and make full use of feature space, we express data using the RBF kernel with different bandwidth [45, 61] based on two facts. On one hand, as a generalized version of linear kernel, RBF kernel is able to map training samples to a higher dimensional feature space. On the other hand, it has fewer parameters than polynomial kernel, leading to the higher computational efficiency. In order to investigate the contribution of each contrast operation, we couple each representation with its corresponding distance function $d_m(\cdot, \cdot)$, and obtain a set of M dissimilarity-based kernels:

$$k_m(f_n, f_{n'}) = \exp\left(\frac{-d_m^2(f_n, f_{n'})}{\sigma_m^2}\right) = \exp\left(\frac{-||f_n - f_{n'}||_2^2}{\sigma_m^2}\right) \tag{15}$$

where σ_m^2 is the bandwidth, always setted as a positive constant, and $||\cdot||_2$ denotes ℓ_2-norm. The usage of dissimilarity-based kernels is advantageous and convenient in solving many visual learning tasks, especially due to the fact that a number of well-designed descriptors and their associated distance functions have been introduced in the literature over recent years [45, 53, 61]. However, k_m in (15) is not always guaranteed to be positive definite. Following [54], we resolve this issue by adding small positive bias term to k_m. Notice that because $\sum_n \alpha_n y_n = 0$ in a standard ℓ_2-norm SVM [30], this bias term is not required in (12). Given (12) and (15), determining a set of optimal ensemble coefficients $\boldsymbol{\beta} = \{\beta_1, \beta_2, ..., \beta_M\}$ can now be interpreted as finding appropriate weights for best fusing different kind of contrast feature maps.

3.3.4 Optimization

Directly optimizing (14) is difficult, we thus resort to an iterative, EM-like strategy to alternately optimize $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, separately. In each iteration, one of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ is optimized while the other is fixed, and then the roles of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are switched. The whole iterations are repeated until convergence is reached. In particular, we will show that the optimization of $\boldsymbol{\beta}$ can be carried out in a closed form.

On optimizing $\boldsymbol{\alpha}$. Suppose we are given the optimized parameter $\boldsymbol{\beta}^*$, the optimization problem of (14) becomes:

$$\begin{aligned}\max_{\boldsymbol{\alpha}} J(\boldsymbol{\alpha}) &= -\frac{1}{2}\sum_{n,n'} \alpha_n \alpha_{n'} y_n y_{n'} \sum_m \beta^*_m k_m(f_n, f_{n'}) \\ &\quad + \sum_n \alpha_n \\ &\quad s.t. \quad \sum_n \alpha_n y_n = 0 \quad 0 \le \alpha_n \le C \quad \forall n \end{aligned} \tag{16}$$

which is identified as the standard SVM dual formulation using the combined kernel $\mathbf{K}(f_n, f) = \sum_m \beta^*_m k_m(f_n, f)$. Thus the objective value $J(\boldsymbol{\alpha})$ can be obtained by any SVM algorithm.

On optimizing $\boldsymbol{\beta}$. Suppose we are given the optimized parameter $\boldsymbol{\alpha}^*$, the optimization problem of (14) becomes:

$$\begin{aligned}\min_{\boldsymbol{\beta}} J(\boldsymbol{\beta}) &= -\frac{1}{2}\sum_{n,n'} \alpha^*_n \alpha^*_{n'} y_n y_{n'} \sum_m \beta_m k_m(f_n, f_{n'}) \\ &\quad + \sum_n \alpha^*_n \\ s.t. \quad & \sum_n \alpha^*_n y_n = 0 \quad 0 \le \alpha^*_n \le C \quad \forall n \\ & \beta_m \ge 0, \quad \sum_m \beta_m = 1 \end{aligned} \tag{17}$$

which is actually a non-linear objective function with constraints over the simplex. With our positivity definition on the kernel functions, $J(\boldsymbol{\beta})$ is convex and differentiable. Thus we solve this problem using a reduced gradient method. By simple differentiation of the objective function of (17) with respect to β_m, we have:

$$\frac{\partial J(\boldsymbol{\beta})}{\partial \beta_m} = -\frac{1}{2}\sum_{n,n'} \alpha^*_n \alpha^*_{n'} y_n y_{n'} k_m(f_n, f_{n'}) \quad \forall n \tag{18}$$

Once the gradient of $J(\boldsymbol{\beta})$ is computed, $\boldsymbol{\beta}$ is updated using a descent direction where the non-negative and normalized constraint of (17) are satisfied. On one hand, we handle the normalized constraint by computing the reduced gradient $\nabla J(\boldsymbol{\beta})$ as follows:

$$\nabla J(\boldsymbol{\beta}) = \begin{cases} \nabla J(\boldsymbol{\beta})_m = \frac{\partial J(\boldsymbol{\beta})}{\partial \beta_m} - \frac{\partial J(\boldsymbol{\beta})}{\partial \beta_\mu}, & \forall m \neq \mu \\ \nabla J(\boldsymbol{\beta})_\mu = \sum_{\mu \neq m} \frac{\partial J(\boldsymbol{\beta})}{\partial \beta_\mu} - \frac{\partial J(\boldsymbol{\beta})}{\partial \beta_m} & \forall m = \mu \end{cases} \tag{19}$$

where β_μ is the largest non-zero component of $\boldsymbol{\beta}$ for better numerical stability as well as [4] does.

On the other hand, we also take into account the positivity constraints of $\boldsymbol{\beta}$ in computing descent direction of (19). Consider the m^{th} entry of $\boldsymbol{\beta}$ where $\beta_m - \nabla J(\boldsymbol{\beta})_m < 0$, this would violate the non-negative constraint for β_m since $-\nabla J(\boldsymbol{\beta})_m$ is a descent direction. Hence, the descent direction for that component is set to 0. This leads to our final descent direction for updating β_m:

$$\nabla J_m = \begin{cases} 0 \qquad \text{if } \beta_m - \left(\frac{\partial J(\boldsymbol{\beta})}{\partial \beta_m} - \frac{\partial J(\boldsymbol{\beta})}{\partial \beta_\mu}\right) < 0 \\ -\frac{\partial J(\boldsymbol{\beta})}{\partial \beta_m} + \frac{\partial J(\boldsymbol{\beta})}{\partial \beta_\mu} \text{ if } \beta_m > 0 \text{ and } m \neq \mu \\ \sum_{m \neq \mu, \beta_m > 0} \left(\frac{\partial J(\boldsymbol{\beta})}{\partial \beta_m} - \frac{\partial J(\boldsymbol{\beta})}{\partial \beta_\mu}\right) \text{ for } m = \mu \end{cases} \tag{20}$$

So far, we update $\boldsymbol{\beta}$ as $\boldsymbol{\beta} \leftarrow \boldsymbol{\beta} + \gamma \nabla J$ and normalize the updated $\boldsymbol{\beta}$, where γ is the step size. As listed in Algorithm 1, the procedure of our method requires an initial guess to $\boldsymbol{\beta}$ in the alternating optimization, where one can start with equal weights for each entry of $\boldsymbol{\beta}$ or other simple initialization such as setting one entry as 1 and remaining as 0. The whole algorithm is terminated when a stopping criterion is achieved. Here a simple stopping criterion is adopted based on the variation of $\boldsymbol{\beta}$ between two consecutive iterative steps. One can use other simple criterion such as maximal number of iterations. Pertaining to the convergence of the whole optimization procedure, the values of the objective function are not guaranteed to monotonically decrease throughout the iterations. Still, the optimization procedures rapidly converge after only a few iterations in our experiments, as shown in Section 4.7.

Algorithm 1 The training procedure of our algorithm

```
Input: Training data: $f_1, f_2, \cdots, f_N$; Associated data label:
        $y_1, y_2, \cdots, y_N \in \{+1, -1\}$;
Initial kernel weights: $\boldsymbol{\beta} = \{\beta_1, \beta_2, \cdots, \beta_M\}$; Initial temp weights: $\boldsymbol{T} = \mathbf{0}$;
Basic kernel constant: $\sigma_m^2$; Step size: $\gamma$;
Stopping parameters: $\varepsilon$;
Output: Model coefficients: $\boldsymbol{\alpha}$; Basic kernel weights (feature map weights): $\boldsymbol{\beta}$
for $||\boldsymbol{T} - \boldsymbol{\beta}||_2 \geq \varepsilon$ do
    Save current $\beta$ as $\boldsymbol{T} = \boldsymbol{\beta}$;
    E-step: Optimize $\boldsymbol{\alpha}^*$
        Compute $\boldsymbol{\alpha}^*$ using a standard SVM solver with fixed $\boldsymbol{\beta}$ and
    $k(f_n, f) = \sum_m \beta_m k_m(f_n, f)$;
    M-step: Optimize $\boldsymbol{\beta}^*$
        Compute descent direction $\nabla J$ for $\boldsymbol{\beta}$ using (20) to satisfy the non-negative
    constraint in (17);
        Update $\boldsymbol{\beta}^*$ as $\boldsymbol{\beta}^* \leftarrow \boldsymbol{\beta} + \gamma \nabla J$;
        Normalize $\boldsymbol{\beta}^*$ to satisfy the equality constraint in (17);
end
return $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$;
```

4 Experimental evaluation

We test our method over four datasets, including TORONTO [7], MIT [31], ASD [10], and SED [6], where the first two are well known for human eye fixation prediction and the later two are popular for salient object detection.

4.1 Experimental setting

4.1.1 Dataset

TORONTO [7] dataset includes 120 color images, which come from indoor and outdoor environments. All images of this dataset are with fixed resolution of 511 × 681. To annotate the ground truth of this dataset, each image is randomly presented to 20 human subjects, where each subject is asked to label each image for 3 seconds with 2 seconds delay in between.

MIT [31] dataset is a larger dataset containing for eye fixation prediction. Compared with TORONTO dataset, it contains 1003 images collected from Flicker and LabelMe datasets, where 779 images are from landscape environment and 228 images are from portrait scenario. The associated ground truth are generated using the eye fixation data collected from 15 human subjects, where each subject are asked to freely view images for 3 seconds with 1 seconds delay in between.

ASD [10] dataset is very popular to evaluate object-based attention models. It contains 1000 color images with pixel-wised ground truth, where each object shape and boundary are well delineated. The image resolutions of this dataset are approximately 400 × 300 or 300 × 400 pixels, and each image only have one salient object.

SED [6] dataset is also used for salient object detection. Compared with ASD dataset, it is a smaller dataset only containing 100 images. Even so, it is more challenging for the task of salient object detection due to two factors: there is no center-biased assumption for salient regions, and each image at least has two salient objects with different scales. Similar to ASD dataset, this dataset also provides accurate object-contour-based ground truth.

4.1.2 Baselines

To show the advantages of our approach, we selected 6 state-of-the-art models as baselines, including spectral residual saliency (SR [21]), attention measure (IT [25]), nature statistic saliency (SUN [68]), multiscale convolutional neural network saliency (MCNN [37]), unified saliency (US [55]), and Co-bootstrapping saliency (CS [24]). Besides, we directly borrow three feature maps (CESC, AWCSC and GC) as baselines for comprehensive comparison. Experimental results of some baseline models are produced using default parameter settings given by the authors, while others are directly borrowed in the literature for comparison. All the baselines are used for the both task of human eye fixation prediction and salient object detection.

4.1.3 Evaluation metrics

We utilize receiver operating characteristic (ROC) curve to evaluate our system. Under this criteria, each predicted saliency map is thresholded to generate the final map. The pixels with larger saliency values than the threshold are identified as salient (positive samples), and the other pixels are considered as non-salient (negative samples) [7]. The ROC curve is plotted with the true positive rate against the false positive rate under varying threshold. After that, we also compute the area under ROC curve (AUC) score for direct comparison.

As discussed in [68] and [56], however, there is always a center bias that our HVS always prefers to the center of an image. Therefore, we turn to the shuffled AUC (sAUC) score [68] as an alternative metric for the task of human eye fixation prediction. In this criteria, sAUC uses all human fixations (except the positive set) as the negative set, instead of selecting negative samples randomly from a uniform distribution. Thus, the center bias can be effectively alleviated.

For the task of salient object detection, we also adopt mean absolute error (MAE) [49] as another evaluation criterion. It is defined as average pixelwise absolute difference between the estimate saliency map and binary ground truth:

$$MAE = \frac{1}{A} \sum_{\mathbf{x}} |S(\mathbf{x}) - G(\mathbf{x})| \tag{21}$$

where A is the area of an input image. MAE measures the numerical dissimilarity between the predicted saliency map and the corresponding ground truth, and is more meaningful in evaluating the applicability of a saliency model for the task of object segmentation.

4.1.4 Implementation details

To train our WLMKL model, we randomly splits all images into two sets: 10% for training, and 90% for testing, over all four datasets. A 10-fold cross validation is employed to reduce the effect of randomly selecting training images, and the results are reported as the average over 10-fold cross validation. The default parameter settings include: neighborhood patch number $L = 8$ in computing CESC and AWCSC, bin number $B = 50$ in computing GC, step size $\gamma = 0.2$, and stopping parameter $\varepsilon = 10^{-4}$. The initialization of each entry of $\boldsymbol{\beta}$ is seted with equal weights. To construct ensemble kernels, we set a series of σ_m^2 as 2, 4, 6, 8, and 10 empirically in (15), following the previous MKL learning problem [8, 45, 57]. The training and testing processes are performed on a 4-core i5 personal computer with 2.6 GHz CPU and 16 GB memory.

To train WLMKL model, we are required to select positive and negative training samples, which identify the associated pixel is salient or not. For the **TORONTO** and **MIT** datasets, it is easy to choose training samples according to the ground truth, where the fixed pixels are identified as positive samples, and non-fixed pixels are treated as negative samples. However, the ground truth of **ASD** and **SED** datasets are binary segmentation masks, where the white pixels denote salient object and black pixels indicate non-salient background. We thus random sample white and black pixels as positive and negative samples to train our WLMKL model.

4.2 Results and analysis

Table 1 shows the performance comparison between the proposed WLMKL method and the baseline methods in terms of the AUC, sAUC and MAE. The corresponding ROC curves are illustrated in Fig. 4. Results show that our WLMKL method outperforms the state-of-the-art approaches. For the task of human eye fixation prediction, our method averagely improves the results with 0.020 and 0.015 in terms of AUC and sAUC, while for the task of salient object detection, our approach achieves better performance with a margin of 0.023 and 0.008 in terms of AUC and MAE, respectively. We also show the promising results of each contrast feature map (AWCSC, GC, and CESC) over four datasets, especially the proposed AWCSC almost gains higher performance than other two contrast measure. In particular,

Table 1 Performance comparison of the baseline methods and our approach on four datasets in terms of AUC, sAUC and MAE

	TORONTO [7]		MIT [31]		ASD [10]		SED [6]	
Methods	AUC	sAUC	AUC	sAUC	AUC	MAE	AUC	MAE
SR [21]	0.516	0.409	0.544	0.437	0.581	0.215	0.538	0.291
IT [25]	0.739	0.627	0.725	0.614	0.826	0.195	0.824	0.271
SUN [68]	0.670	0.505	0.722	0.609	0.739	0.283	0.780	0.342
US [55]	0.534	0.447	0.515	0.422	0.861	0.205	0.828	0.323
CS [24]	0.815	0.670	0.804	0.658	0.827	0.219	0.803	0.305
MCNN [37]	0.817	0.659	0.814	0.656	0.754	0.237	0.737	0.337
CESC	0.691	0.671	0.677	0.603	0.659	0.157	0.796	0.264
GC	0.816	0.690	0.808	0.676	0.791	0.171	0.850	0.258
AWCSC	0.811	0.694	0.816	0.670	0.828	0.153	0.861	0.252
Ours	0.827	0.702	0.843	0.697	0.902	0.146	0.876	0.244

on the SED dataset, AWCSC achieves an AUC of 0.861 and a MAE value of 0.252, and outperforms the best performing baselines with a margin of 0.015 and 0.008, respectively.

Among all datasets, our method achieves less improvement on the SED dataset, where the contained images have more than one salient object. Such kind of factors bring difficulties for the saliency detection. Even so, our WLMKL still has outstanding performance benefiting from its capability of learning the optimized combination of contrast feature maps for salient object detection. It is interesting that our method works better on MIT and ASD datasets, which indicate our method has good scalability over large datasets.

Some examples of the saliency maps produced from our WLMKL and the baseline methods are shown in Fig. 5. One can observe that WLMKL produces saliency maps more consistent with the ground truth, compared with other baselines. These results clearly demonstrate the effectiveness of WLMKL in combining the contrast feature maps to perform visual saliency detection. It is worth noting that the proposed WLMKL does not require the preprocessing such as over-segmentation, nor any assistance from the detection of objects.

4.3 Cross data evaluation

In order to investigate the generalization ability of our WLMKL model, we perform experiment in the scenario of cross dataset evaluation, and compare with a baseline approach [5]

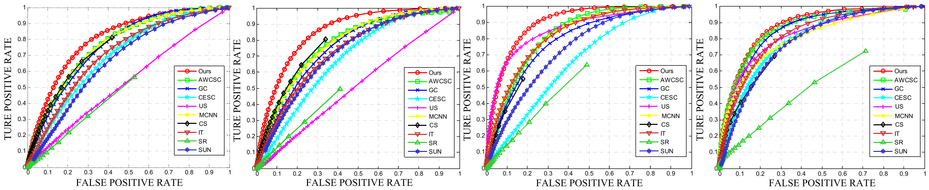


Fig. 4 ROC curve comparison between our method and other basline approaches. From left to right are the results on the TORONTO, MIT, ASD and SED datasets. (Best viewed in color)

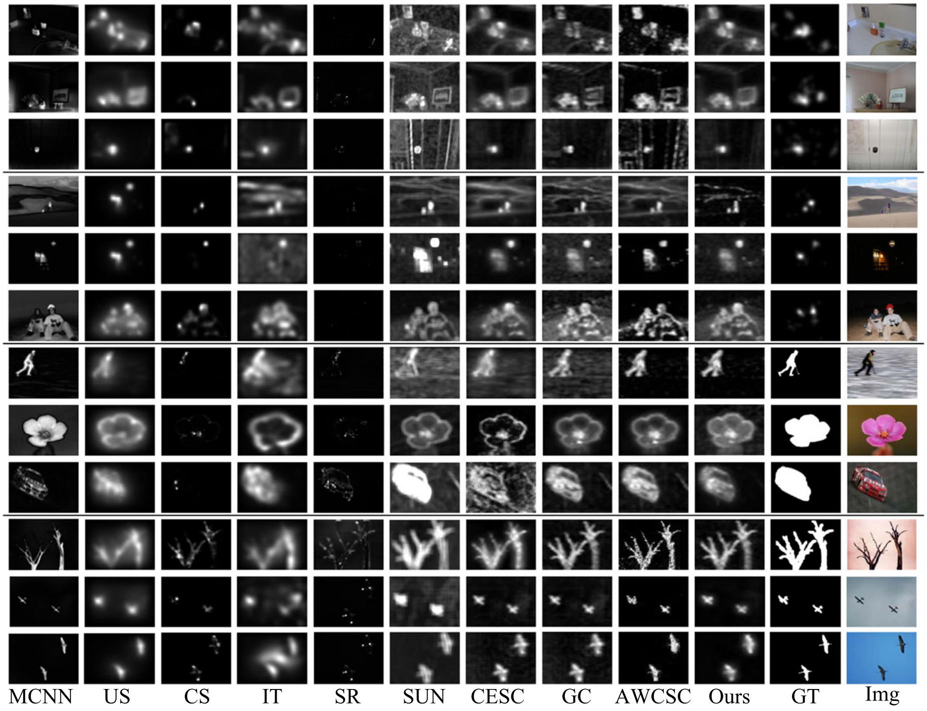


Fig. 5 Visual comparison between our method and other basline approaches. From top to bottom are some examples of predicted saliency maps on the TORONTO, MIT, ASD and SED datasets. The columns from left to right, respectively, show estimated saliency maps produced by MCNN, US, CS, IT, SR, SUN, CESC, GC, AWCSC, and our proposed methods, with corresponding ground truth and original images. (Best viewed in color)

that also learns feature weights via boosting model (FWBL). More specifically, we still randomly sample 10% images from one dataset for training, and then apply the trained model to all the images from the remaining datasets for testing. Table 2 reports the saliency detection results.

From Table 2, we observe that our method is consistently superior to FWBL. The performance of WLMKL only have a negligible drop when it is applied on different datasets

Table 2 Cross dataset evaluation between our WLMKL model and FWBL [5] in terms of AUC

Training dataset	Testing dataset	AUC	
		WLMKL	FWBL
TORONTO	TORONTO MIT ASD SED	0.827 0.784 0.717 0.709	0.808 0.767 0.694 0.678
MIT	TORONTO MIT ASD SED	0.811 0.843 0.753 0.718	0.752 0.773 0.681 0.647
ASD	TORONTO MIT ASD SED	0.845 0.812 0.902 0.867	0.803 0.761 0.886 0.854
SED	TORONTO MIT ASD SED	0.814 0.803 0.855 0.876	0.770 0.756 0.838 0.855

within the same saliency detection task. For human eye fixation predction, when learning the optimized feature combinations and testing on TORONTO dataset, WLMKL achieves a AUC score of 0.827. When learning the optimized feature combinations on MIT dataset and testing on TORONTO dataset, we achieve a AUC score of 0.811. The performance only drops 0.016. In contrast, with the same setting, the performance of FWBL [5] drops as large as 0.056. We also observe the similar results for the task of salient object detection. These results well demonstrate that the our WLMKL model offers a good generalization ability for different datasets.

We also investigate the performance of our WLMKL model for different saliency detection task. The results in Table 2 show that the performance significantly drops within different visual saliency detection task, specifically for detecting the salient object. This is probably because, compared with fixation prediction, salient object segmentation provides different ground truth, which requires object level segmentation with clear boundary declination. However, the learned WLMKL from fixation datasets tends to produce scattered fixation estimation. This distinctness inevitably makes WLMKL perform worse within different visual saliency detection task. This conclusion is also consistent with the observations of [66]. The model for one saliency detection task hardly achieves good performance for the other task due to different priors provided from the ground truth.

4.4 Analysis of individual contrast feature maps

In order to quantize the contribution of each subcomponent of our approach, we evaluate the effect of sequentially inducing individual contrast feature map, and analysis their contribution for final saliency maps. To be specific, we first only use CESC, AWCSC, and GC to train our WLMKL model as baselines, then different combinations of feature maps are evaluated in terms of AUC. Specifically, we sum up the learned weights of each contrast feature among all kernels, and the results are list in Table 3.

It is clear that the combination is superior than individual feature. Additionally, combined CESC with GC, or CESC with AWCSC, or GC with AWCSC obtain comparable results. It is also observed that the simple using of AWCSC appears to be surprisingly effective on four datasets, achieving 0.821, 0.828, 0.847 and 0.863, respectively. Compared with Tables 1 and 2, in spite of only using AWCSC to train our model, it still outperforms some baseline methods. Another interesting conclusion is that although employing individual CESC to perform training is ranked at the bottom, it significantly improves the results when combined with GC feature map. Figure 6 shows some visual examples of saliency maps on four datasets, comparing the individual contribution of induced contrast feature maps and the overall results, respectively.

Table 3 Contributions of different feature maps and their combinations to the performance in terms of AUC

	TORONTO	MIT	ASD	SED
CESC	0.734	0.742	0.738	0.811
GC	0.819	0.819	0.803	0.852
AWCSC	0.821	0.828	0.847	0.863
CESC+GC	0.822	0.834	0.848	0.865
CESC+AWCSC	0.823	0.833	0.861	0.868
AWCSC+GC	0.825	0.836	0.884	0.872
AWCSC+CESC+GC	0.827	0.843	0.902	0.876

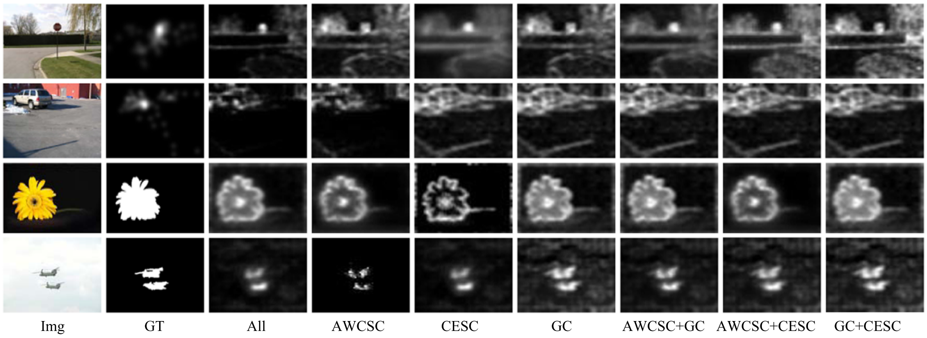


Fig. 6 Some visual examples of saliency maps to illustrate individual subcomponents and their integrations. From up to bottom are samples from TORONTO, MIT, ASD and SED datasets. (Best viewed in color)

From Table 3, we can also conclude that, when combined with CESC, AWCSC, and GC feature maps, the performance can be averagely improved by 0.011, 0.045, and 0.039, respectively. This indicates our AWCSC always plays an important role in the production of final saliency map on four datasets. To further understand the contribution of individual feature map, we also list the learned weights for each subcomponent over four datasets in Table 4. Once again, one can observe that AWCSC is assigned the largest feature weight, which is consistent with the conclusion of Table 3.

4.5 Analysis of parameter settings

Two factors directly affecting the performance are the neighborhood patch number L associated in the calculation of CESC and AWCSC , and the bin number B in the computation of GC. In Fig. 7, we thus evaluate our method by changing the values of these two parameters, using all the available contrast feature maps (CESC, AWCSC, and GC).

As shown in the left panel of Fig. 7, increasing the number of neighbors reduces the accuracy of each individual contrast operator. Correspondingly, the performance of our WLMKL model is also dropped. The right diagram of Fig. 7 shows that the AUC score of our method peaks at approximately 50 bins over four datasets, and significant drops when $B = 100$. After that, any refinement to this parameter will result in slightly fluctuation of performance. This is probably because that changing the bin number will result in poor prediction of GC feature map, together with this contrast feature has secondary contribution to the estimation of final saliency map (as shown in Table 4), leading to our approach performs worse when B is around 100.

Table 4 Learning weights for different datasets

	TORONTO	MIT	ASD	SED
CESC	0.122	0.031	0.149	0.114
AWCSC	0.446	0.507	0.491	0.481
GC	0.432	0.462	0.360	0.405

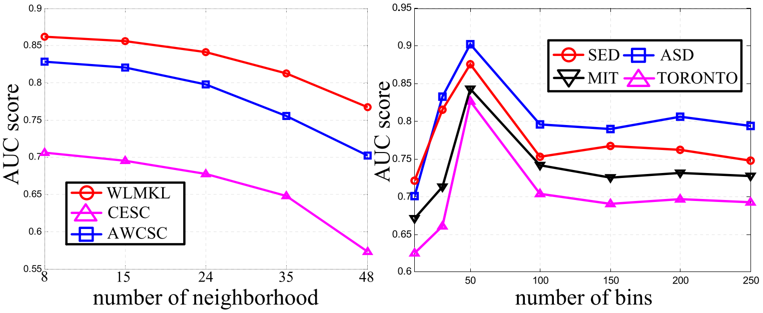


Fig. 7 Illustration of parameter settings. Left: Average AUC score along with increasing the number of surround neighborhoods L over four datasets. Right: Effect of the performance with different bin number B on four datasets. (Best viewed in color)

4.6 Analysis of implemental efficiency

In order to evaluate the implemental efficiency of our WLMKL model, we compare the average running time per image with the baseline models, and report the results in Table 5. Our method is faster than IT [25] and US [55], and slower than other baseline approaches. The proposed WLMKL model, however, is still running fast, producing a saliency map of one test image no more than a second on a conventional personal computer. The bulk of the computation resides in the construction of contrast feature maps (about 80%), and only 20% account for the actual saliency calculation. Once trained, our WLMKL model is parameter free and requires no adjustment of thresholds or other knobs.

4.7 Analysis of convergence

In Fig. 8, we also analysis the convergence of the optimization algorithm via the value of learned feature weights vs. the number of iterations. It demonstrates that our algorithm rapid converges to the stable value among AWCSC, GC, and CESC, respectively. Precisely, in TORONTO and SED datasets, our approach converges at nearly 40 iterations, while for MIT and ASD datasets, the learned values have no significant changes after 50 iterations. It is worth noticing that, compared with other two datasets, our method converges faster in

Table 5 Average running time comparison per image on four datasets

	IT [25]	US [55]	CS [24]	MCNN [37]	SR [21]	SUN [68]	Ours
TORONTO	1.772	4.680	0.052	0.046	0.186	0.093	0.687
MIT	2.261	5.578	0.062	0.059	0.221	0.111	0.896
ASE	0.611	1.614	0.018	0.016	0.064	0.032	0.375
SED	0.344	1.029	0.011	0.013	0.041	0.039	0.214
Code	Matlab	C++	C++	C++	Matlab	Matlab	Matlab

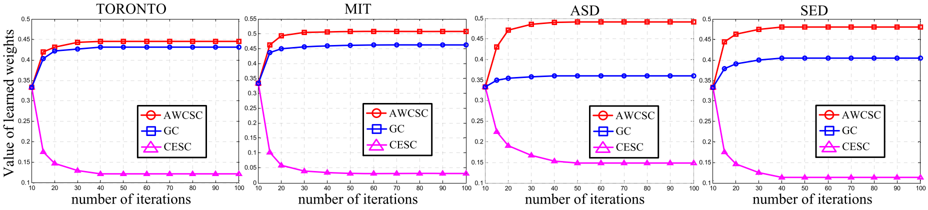


Fig. 8 Illustration of the convergence of our method. From left to right are the summation of learned feature weights for each contrast feature map along with increasing iteration number over four datasets. (Best viewed in color)

TORONTO and SED datasets, the main reason probably lies in that these two datasets has smaller number of images, leading to less training data used to learn our WKMKL model.

5 Conclusions and future work

In this paper, a WLMKL framework is proposed for visual saliency detection. WLMKL learns adaptive weights to incorporate three contrast feature maps, namely, AWCSC, CESC and GC, respectively. Our WLMKL model enables each contrast feature map contributes to predict pixel saliency via preserving salient features and suppressesing the nonsalient features. We evaluate our method for two visual attention task: human eye fixation estimation and salient object detection. Extensive experiments well validate the effectiveness of our framework on four benchmark datasets.

In the future, there are two areas that we would like to improve upon. The first one is investigating more perceptual color space jointly to further improve the performance, as well as [13] does. Secondly, we would like to explore more feature space (e.g., texture feature and edge strength) to further enhance performance, and apply our method for other visual tasks, such as image construction [40] and medical image segmentation [38, 39].

Acknowledgements The authors would like to thank all the anonymous reviewers for their valuable comments and suggestions. This work was partly supported by the National Science Foundation (Grant No. IIS-1302164), the National Natural Science Foundation of China (Grant No. 61876093, 61881240048, 61671253, 61701252, 61762021), Natural Science Foundation of Jiangsu Province (Grant No. BK20181393, BK20150849, BK20160908), Huawei Innovation Research Program (HIRP2018), and Natural Science Foundation of Guizhou Province (Grant No.[2017]1130).

References

1. Achanta R et al (2009) Frequency-tuned salient region detection. In: CVPR, pp 1597–1604
2. Alexe B, Deselaers T, Ferrari V (2010) What is an object? In: CVPR, pp 73–80

3. Bach FR et al (2004) Multiple kernel learning, conic duality, and the SMO algorithm. In: ICML, pp 6–14
4. Bonnans F (2006) Optimisation continue. Dunod
5. Borji A (2012) Boosting bottom-up and top-down visual features for saliency estimation. In: CVPR, pp 438–445
6. Borji A et al (2015) Salient object detection: a benchmark. TIP 24(12):5706–5722
7. Bruce N, Tsotsos J (2006) Saliency based on information maximization. In: NIPS, pp 155–162
8. Bucak S et al (2014) Multiple kernel learning for visual object recognition: a review. TIP 36(7):1354–1369
9. Chang KY et al (2011) Fusing generic objectness and visual saliency for salient object detection. In: ICCV, pp 914–921
10. Cheng MM et al (2011) Global contrast based salient region detection. In: CVPR, pp 409–416
11. Cornia M et al (2016) A deep multi-level network for saliency prediction. In: ICPR, pp 3488–3493
12. Duan L et al (2011) Visual saliency detection by spatially weighted dissimilarity
13. Fernandez-Carbajales V et al (2016) Visual attention based on a joint perceptual space of color and brightness for improved video tracking. Pattern Recogn 60:571–584
14. Fu Y et al (2008) Saliency cuts: an automatic approach to object segmentation. In: ICPR, pp 1–4
15. Gao DS et al (2008) On the plausibility of the discriminant center-surround hypothesis for visual saliency. J Vis 8(7):13–25
16. Gao DH, Han S, Vasconcelos N (2009) Discriminant saliency, the detection of suspicious coincidences, and applications to visual recognition. TPAMI 31(6):989–1005
17. Goferman S et al (2012) Context-aware saliency detection. TPAMI 34(10):1915–1926
18. Gopalakrishnan V et al (2009) Salient region detection by modeling distributions of color and orientation. TMM 11(5):892–905
19. Guo C, Zhang L (2010) A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression. TIP 19(1):185–198
20. Harel J et al (2006) Graph-based visual saliency. In: NIPS, pp 545–552
21. Hou X, Zhang L (2007) Saliency detection: a spectral residual approach. In: CVPR, pp 1–8
22. Hou X, Zhang L (2008) Dynamic visual attention: searching for coding length increments. In: NIPS, pp 681–688
23. Huang X et al (2015) SALICON: reducing the semantic gap in saliency prediction by adapting deep neural networks. In: ICCV, pp 262–270
24. Huchuan L et al (2017) Co-bootstrapping saliency. TIP 26(1):414–425
25. Itti L (1998) Others: a model of saliency-based visual attention for rapid scene analysis. TPAMI 20(11):1254–1259
26. Itti L, Koch C (2001) Computational modelling of visual attention. Nat Rev Neurosci 2(3):194–203
27. Jetley S et al (2016) End-to-end saliency mapping via probability distribution prediction. In: CVPR, pp 5753–5761
28. Jing PG, Su YT, Nie LQ, Bai X, Liu J, Wang M (2018) Low-rank multi-view embedding learning for micro-video popularity prediction. TKDE 30(8):1519–1532
29. Jing PG, Su YT, Nie LQ, Gu HM, Liu J, Wang M (2018) A framework of joint low-rank and sparse regression for image memorability prediction CSVT. https://doi.org/10.1109/TCSVT.2018.2832095
30. John ST, Cristianini N (2004) Kernel methods for pattern analysis. Cambridge University Press, Cambridge
31. Judd T et al (2009) Learning to predict where humans look. In: ICCV, pp 2106–2113
32. Kadir T, Brady M (2001) Saliency, scale and image description. IJCV 45(2):83–105
33. Koch C, Ullman S (1985) Shifts in selective visual attention: towards the underlying neural circuitry. Matters Int 4(1):219–227
34. Kruthiventi SSS et al (2017) Deepfix: a fully convolutional neural network for predicting human eye fixations. TIP 26(9):4446–4456
35. Kummerer M et al (2015) Deep gaze I: boosting saliency prediction with feature maps trained on ImageNet. In: ICLRW, pp 262–270
36. Liu N, Han JJ (2016) DHSNet: Deep hierarchical saliency network for salient object detection. In: CVPR, pp 678–686
37. Liu W et al (2018) Learning to predict eye fixations via multiresolution convolutional neural networks. TNNLS 29(2):392–404
38. Lu H et al (2017) Wound intensity correction and segmentation with convolutional neural networks. Concurrency and Computation: Practice and Experience 29(6):3927–3737

39. Lu H et al (2018) Brain intelligence: go beyond artificial intelligence. Mobile Netw Appl 23(2):368–375
40. Lu H et al (2018) Low illumination underwater light field images reconstruction using deep convolutional neural networks. Futur Gener Comput Syst 82(6):142–148
41. Ma YF, Zhang HJ (2003) Contrast-based image attention analysis by using fuzzy growing. In: ACMMM, pp 374–381
42. Ma Q, Zhang L (2008) Image quality assessment with visual attention. In: ICPR, pp 1–4
43. Mairal J et al (2010) Online learning for matrix factorization and sparse coding. J Mach Learn Res 11(1):19–60
44. Marchesotti L et al (2009) A framework for visual saliency detection with applications to image thumbnailing. In: ICCV, pp 2232–2239
45. Mehmet G et al (2011) Multiple kernel learning algorithms. JMLR 12(7):2211–2268
46. Nuthmann A, Henderson JM (2010) Object-based attentional selection in scene viewing. J Vis 10(8):2237–2242
47. Olshausen BA et al (1996) Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature 381(7):607–609
48. Pan J et al (2016) Shallow and deep convolutional networks for saliency prediction. In: CVPR, pp 598–606
49. Perazzi F et al (2012) Saliency filters: contrast based filtering for salient region detection. In: CVPR, pp 733–740
50. Quan Z et al (2018) Weighted linear multiple kernel learning for saliency detection. In: ROSENET
51. Shen X, Wu Y (2012) A unified approach to salient object detection via low rank matrix recovery. In: CVPR, pp 853–860
52. Simoncelli EP et al (2001) Natural image statistics and neural representation. Annu Rev Neurosci 24(1):1193–1216
53. Sonnenburg S et al (2006) Large scale multiple kernel learning. JMLR 7(1):1531–1565
54. Sonnenburg S et al (2012) Object detection with dog scale-space: a multiple kernel learning approach. TIP 21(8):3744–3756
55. Srinivas SS et al (2016) Saliency unified: a deep architecture for simultaneous eye fixation prediction and salient object segmentation. In: CVPR, pp 5781–5790
56. Tatler BW et al (2015) Visual correlates of fixation selection: effects of scale and time. Vis Res 45(5):643–659
57. Thiagarajan J et al (2014) Multiple kernel sparse representations for supervised and unsupervised learning. TIP 23(7):2905–2915
58. Tian H, Others (2014) Salient region detection by fusing bottom-up and top-down features extracted from a single image. TIP 23(10):4389–4398
59. Torralba A et al (2003) Modeling global scene factors in attention. JOSA A 20(7):1407–1418
60. Treisman AM, Gelade G (1980) A feature-integration theory of attention. Cogn Psychol 12(1):97–136
61. Varma M, Ray D (2007) Learning the discriminative power-invariance trade-off. In: ICCV, pp 1–8
62. Vladimir V (1993) The nature of statistical learning theory. Springer, Berlin
63. Wang LJ, Lu HC, Wang YF, Feng MY, Wang D, Yin BC, Ruan X (2017) Learning to detect salient objects with image-level supervision. In: CVPR, pp 3796–3805
64. Wang W et al (2018) Video salient object detection via fully convolutional networks. TIP 27(1):38–49
65. Xu ZL et al (2010) Simple and efficient multiple kernel learning by group lasso. In: ICML, pp 1175–1182
66. Yin L et al (2014) The secrets of salient object segmentation. In: CVPR, pp 280–287
67. Yu JG et al (2016) A computational model for object-based visual saliency: spreading attention along gestalt cues. TMM 18(2):273–286
68. Zhang LY et al (2008) SUN: A Bayesian framework for saliency using natural statistics. J Vis 8(7):32–42
69. Zhang PP, Wang D, Lu HC, Wang HY, Ruan X (2017) Amulet: aggregating multi-level convolutional features for salient object detection. In: ICCV, pp 202–211
70. Zhang PP, Wang D, Lu HC, Wang HY, Yin BC (2017) Learning uncertain convolutional features for accurate saliency detection. In: ICCV, pp 212–221
71. Zhou Q et al (2013) On contrast combinations for visual saliency detection. In: ICIP, pp 2665–2669
72. Zhou Q et al (2014) Salient object detection using window mask transferring with multi-layer background contrast. In: ACCV, pp 221–235

Quan Zhou He received the M.S. degree and Ph.D. degree in communication and information system in 2006 and 2013, respectively, from Huazhong University of Science and Technology, China. He is now an associate professor of Nanjing University of Posts and Telecommunications. His research interests include computer vision and pattern recognition. He has published over 30 research papers in SCI journals (e.g., IEEE Transactions on Image Processing, IEEE Transactions on Multimedia, Pattern Recognition) and conference (ICIP, ICASSP, ACCV, and ICPR) in image processing and computer vision. He now serves as TPC member or chair of many international conferences and reviewer for a series of SCI journals, including IEEE Transactions on Image Processing, IEEE Transactions on Multimedia, IEEE Transactions on Circuits System for Video Technology, Pattern Recognition, and Neurocomputing. He is member of IEEE.

Jie Cheng He received Ph.D degree in electronic and electricity engineering in 2012, from Huazhong University of Science and Technology, China. He is now the engineering centre director of Huawei Technologies Co. Ltd., China. His research interests include signal processing and image processing.

Huimin Lu He received M.S. degrees in electrical engineering from the Kyushu Institute of Technology, Kitakyushu, Japan, and Yangzhou University, Yangzhou, China, in 2011, and the Ph.D. degree in electrical engineering from the Kyushu Institute of Technology, in 2014. From 2013 to 2016, he was a JSPS Research Fellow with the Kyushu Institute of Technology. He is currently an Assistant Professor with the Kyushu Institute of Technology, Kitakyushu, Japan, and an Excellent Young Researcher of MEXT-Japan. His current research interests include computer vision, robotics, artificial intelligence, and ocean observing.

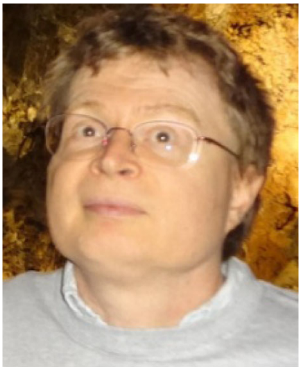
Yawen Fan She received BE and MS degrees in electronic engineering from Hohai University, Nanjing, China in 2003 and 2006, respectively. She has received the doctor degree in EE department from Shanghai Jiao Tong University, Shanghai, China. She is now an assistant professor at Nanjing University of Posts and Telecommunications, Nanjing, P. R. China. Her research interests are intelligent video surveillance and video analysis and understanding.

Suofei Zhang He received the PhD degree in School of Information Science and Engineering from Southeast University in 2013 and the master degree in School of Mechanical Engineering from Jiangsu University in 2007. In 2013, he joined the School of Internet of Things at the Nanjing University of Posts and Telecommunications. From 2009 to 2010 he visited the robots laboratory of ENSTA Paris Tech as researcher. His research interests include computer vision, video surveillance, real-time object tracking and deep learning based image processing.

Xiaofu Wu He received the B.S. and M.S. degrees in electrical engineering from the Nanjing Institute of Communications Engineering, Nanjing, China, in 1996 and 1999, respectively, and Ph. D. degree in electrical engineering from Peking University, Beijing, China, in 2005. From 2005 to 2007, he has with the Southeast University as a Post-Doctoral reseacher at the National Mobile Communication Research Laboratory. Since 2012, he has been with the Nanjing University of Posts and Telecommunications, where he is currently a full Professor. His research interests are in coding and information theory, information-theoretic security, machine learning and computer vision.

Baoyu Zheng He received the B.S. degree in Electronic Science and Technology from the Nanjing University of Posts and Telecommunications (NUPT), Nanjing, China. He is currently a full professor at the College of Telecommunications and Information Engineering, NUPT, China. His research interests include signal processing in communications and quantum signal processing. He also served as chair for communication theory and signal processing of China communication academy, associate chair of Nanjing Communication chapter of IEEE. He is senior member of IEEE.

Weihua Ou He received the M.S. degree in Mathematics from the Southeast University, Nanjing, China in 2006 and the Ph.D. degree in Information and Communication Engineering from Huazhong University of Science and Technology (HUST), China in 2014, respectively. Currently, he is an Associate Professor at the School of Big data and Computer Science in Guizhou Normal University, Guiyang, China. His current research interests include sparse representation, multi-view learning, and image processing and computer vision.



Longin Jan Latecki He received the PhD degree in computer science from Hamburg University, Germany, in 1992. He is a professor of computer science at Temple University, Philadelphia. His main research interests include shape representation and similarity, object detection and recognition in images, robot perception, machine learning, and digital geometry. He has published over 230 research papers and books. He is an editorial board member of Pattern Recognition, Computer Vision and Image Understanding and the International Journal of Mathematical Imaging. He received the annual Pattern Recognition Society Award, together with Azriel Rosenfeld, for the best article published in the journal Pattern Recognition in 1998. He is the recipient of the 2000 Olympus Prize, the main annual award from the German Society for Pattern Recognition (DAGM). He is a senior member of the IEEE.

Affiliations

Quan Zhou[1,2] · Jie Cheng[3] · Huimin Lu[4] · Yawen Fan[1] · Suofei Zhang[5] · Xiaofu Wu[1] · Baoyu Zheng[1] · Weihua Ou[6] · Longin Jan Latecki[7]

Jie Cheng
jiecheng2009@gmail.com

Huimin Lu
luhuimin@ieee.org

Yawen Fan
ywfan@njupt.edu.cn

Xiaofu Wu
xfuwu@njupt.edu.cn

Baoyu Zheng
zby@njupt.edu.cn

Weihua Ou
ouweihuahust@gmail.com

Longin Jan Latecki
latecki@temple.edu

[1] National Engineering Research Center of Communications and Networking, Nanjing University of Posts & Telecommunications, Nanjing, People's Republic of China

[2] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, People's Republic of China

[3] Huawei Technologies Co. Ltd., ShenZhen, People's Republic of China

[4] Department of Mechanical and Control Engineering, Kyushu Institute of Technology, Kitakyushu, Japan

[5] School of Internet of Things, Nanjing University of Posts & Telecommunications, Nanjing, People's Republic of China

[6] School of Big Data and Computer Science, Guizhou Normal University, Guiyang, People's Republic of China

[7] Department of Computer and Information Sciences, Temple University, Philadelphia, USA