Multimedia Tools and Applications (2022) 81:21741–21777
https://doi.org/10.1007/s11042-022-12567-y

Performance evaluation of salient object detection techniques

Kareem Ahmed[1] · Mai A. Gad[1] · Amal Elsayed Aboutabl[2]

Received: 25 March 2021 / Revised: 15 June 2021 / Accepted: 31 January 2022 /
Published online: 16 March 2022

Abstract

Recently, the detection and segmentation of salient objects that attract the attention of human visual in images is determined by using salient object detection (SOD) techniques. As an essential computer vision problem, SOD has increasingly attracted the researchers' interest over the years. While a lot of SOD models and applications have been proposed, there is still a lack of deep understanding of the issues and achievements. A comprehensive study on the recent techniques of SOD is provided in this paper. Precisely, this paper presents a review of SOD techniques from various perspectives. Various image segmentation techniques are presented such as segmentation based on machine learning or deep learning, the second perspective concentrates on classifying them into supervised and unsupervised learning techniques and the last one based on manual approach, semi-automatic approach, and fully automatic approach and so on. Then, the paper presents a summarization of datasets used for SOD. Finally, analyses of SOD models and comparison results are presented.

Keywords Image Saliency · Image Segmentation · Salient Object Detection (SOD) · Machine learning · deep Learning · Medical Image · Remote Sensing image

✉ Kareem Ahmed
kareem_ahmed@hotmail.co.uk

Mai A. Gad
maialaa@fcis.bsu.edu.eg

Amal Elsayed Aboutabl
amal.aboutabl@fci.helwan.edu.eg

[1] Computer Science department, Faculty of Computers and Artificial Intelligence, Beni-Suef University, Beni Suef, Egypt

[2] Computer Science department, Faculty of Computers and Artificial Intelligence, Helwan University, Helwan, Egypt

1 Introduction

The first stage in image classification models is the detection and segmentation of salient objects [1]. As known that humans have an amazing ability to visually determine the salient objects (attention center) accurately and quickly than any machine [2]. For a machine to solve this problem, salient object detection (SOD) is used. The importance of SOD in computer vision applications lies in its ability to reduce the degree of computational complexity. SOD in computer vision is interpreted as a process of two stages: detecting the salient object in the image and segmenting the region of the object respectively [3]. The SOD has been used on a large scale as a preprocessing stage in computer vision applications such as image understanding, object detection and semantic segmentation [4]. Recently, image community spread very rapidly such as Instagram application. Not only the image but also the video material which is basically based on image in every single frame. So, image recognition field has become the milestone for all researchers in this field, especially SOD. The artificial intelligence methods which are used in medical imaging segmentation have been divided into two kinds; the first one is the traditional machine learning techniques, and the second is the deep learning techniques [5]. According to machine learning, the algorithms of medical image segmentation was classified into two types: supervised and unsupervised learning [6]. Also, segmentation techniques are divided into three categories: Manual approach, semi-automatic approach, and fully automatic approach [7]. However, on MRI and CT medical data, segmentation methods grouped into four classes. These classes are region growing, active contour, edge detection, and hybrid techniques [8, 9].

Researchers have conducted several studies in a lot of fields such as medical diagnosis, remote sensing images, Agriculture field, transportation field and so on to identify different targets automatically to reduce errors caused by human and save effort and time [10]. Object detection has a great importance in medical diagnosis of many diseases such as Prostate Cancer [11], Breast Cancer [12], bone diseases [13], and teeth disease [14]. Also, SOD has attracted researchers' attention in remote sensing field; Variations in target size and kind, wide range, vertical views, and Complex backgrounds make the object detection process a challenging task. In the agriculture field, the diseases of the plant cause huge damage, leading to significant crop losses.

The SOD deep learning models offer a robust tool and accurate results for the detection of plant disease. Intelligent Transportations systems can play a significant role in developing the transportation field. The desired information can be extracted from the cameras by the different techniques of artificial intelligence. Samples of salient object detection from many references are presented in In Fig. 1. There are a lot of popular datasets which are used for SOD such as DUTS [15], SBU [16], ISTD [17], UCF [18], SOC [19], DUT-OMRON [20], HKU-IS [21], ECSSD [22], PASCAL-S [23], and SOD [24].

The models which are used for salient object detection are divided into two types; feature-based models and deep learning models as illustrated in Fig. 2. Feature-Based Models are the traditional methods of salient object detection which are considered saliency cues. The contrast is one of them, which analyzes the uniqueness of each patch or pixel in the image regarding local or global contexts. But these methods fail to retain image details and they are not capable of locating salient objects of very large size [30]. Traditional methods can generally be categorized in two ways based on the types of features they exploit [3].

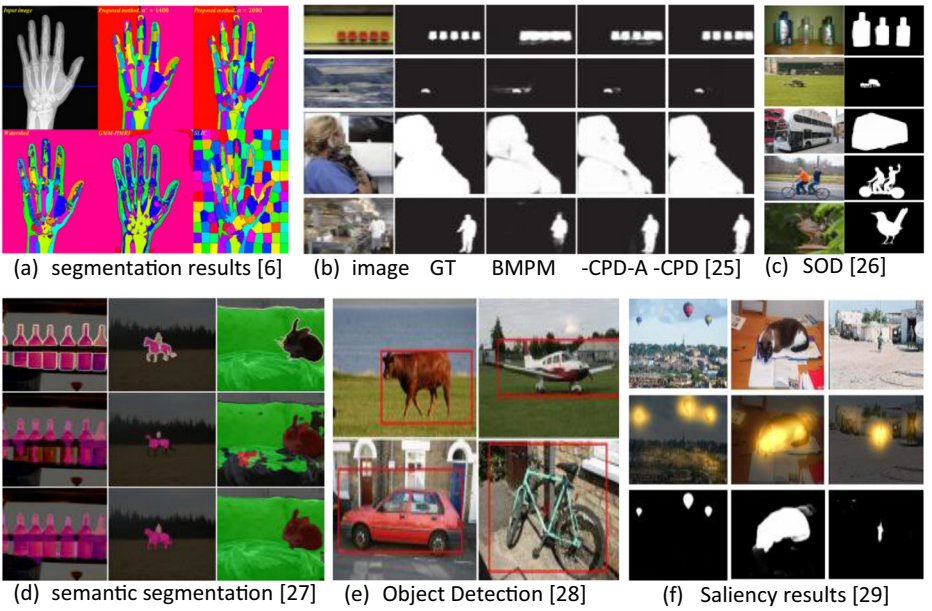

Fig. 1 Salient object detection samples from corresponding references. (a) segmentation results [6], (b) image GT BMPM-CPD-A -CPD [25], (c) SOD [26], (d) semantic segmentation [27], (e) Object Detection [28], (f) Saliency results [29]

Region-based vs. Block based analysis Block based (i.e., patches and pixels) is an early technique of detecting a salient object, while regions are spread out on a large scale with the super-pixel algorithms development [31].

Extrinsic cues vs. intrinsic cues The difference between intrinsic and extrinsic cues is represented in the use of attributes. When using attributes from single image, this is called intrinsic cues. But, in the case of a collaboration of similar images (e.g. Depth map, statistical

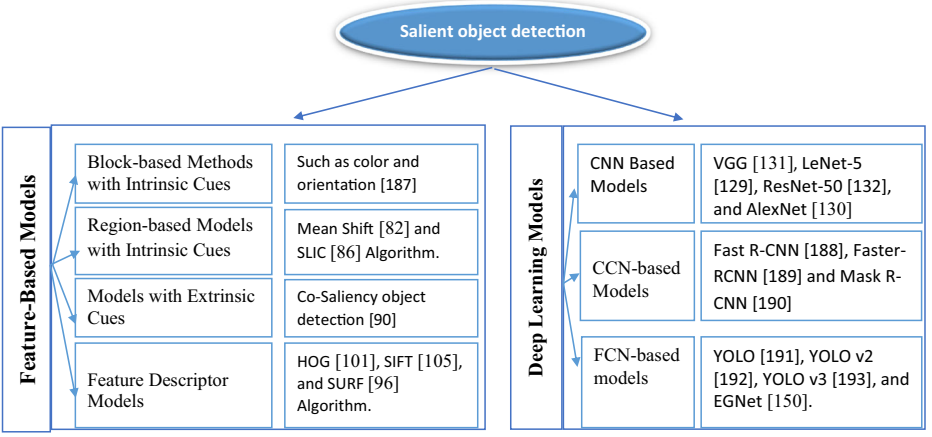

Fig. 2 Classification of salient object detection models

information, or user annotations) to make the detection of salient object in the image easier, this is called extrinsic cues.

Deep learning-based models is one of the most significant techniques that has contributed to the improvement of salient object detection. Recently, many researchers have tended to use deep learning-based techniques to solve the SOD problems, which greatly improves the performance. Convolutional Neural Networks (CNNs) is one of the most used deep learning models and are discussed in detail. [32, 33]

After the detection of salient objects in the image and feature extraction, classification models are used to determine the class label of these objects such as, Random Forest, SVM, logistic regression, KNN and deep learning models. Random Forest (RF) is a Tree-based method for making decisions based on characteristics of multiple Decision Trees. The following techniques use RF for classification [34–38]. The Pseudo Code of RF is shown below, and Fig. 3 illustrates the idea of the algorithm [39].

Random Forest pseudo-Code

1- Select "m" samples from total "k" samples, where m<<k.
2- Among the "m" samples, calculate the node "n" using the best split point.
3- Using the best split, divide the node into daughter nodes.
4- Repeating steps 1 to 3 until "X" nodes number has reached.
5- Repeating steps 1 to 4 for "t" times to build forest and create "t" number of trees.
6- predict the outcome of each decision tree.
7- For each predicted target, voting will be performed and select the most voted prediction result.

Support Vector Machine (SVM) is a non-parametric supervised machine learning model for classification and regression. There are two types of SVM, Linear SVM is the first type which is used for dataset that can be classified using a single straight line into two classes. The second type is Non-linear SVM which is used for datasets that cannot be separated by straight line.

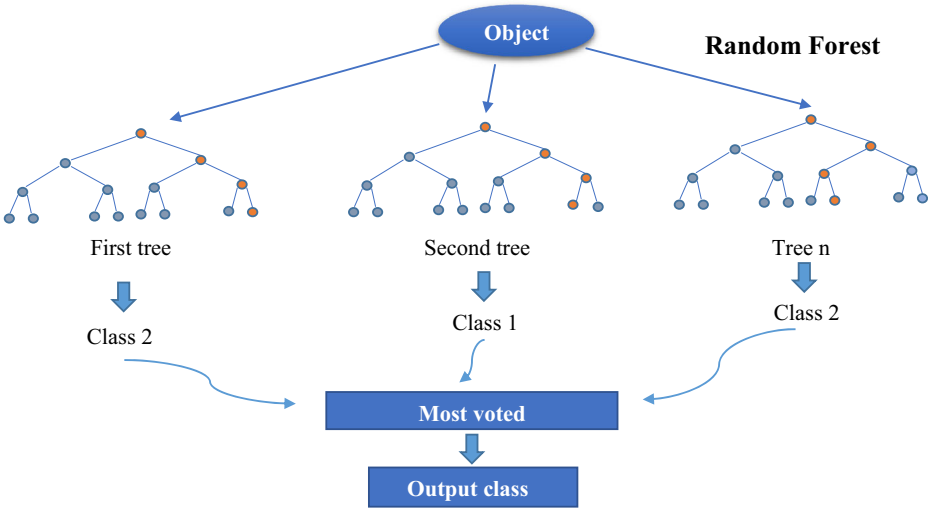

Fig. 3 Simplification of random forest algorithm

The following techniques use SVM for classification [34, 40, 38, 41]. The illustration of SVM algorithm is shown in Fig. 4 [42].

The objective is the selection of maximum marginal hyperplane, and it is selected in the following steps:

1- Generate hyperplanes which separate the classes in the best way.
2- Select the right hyperplane with the max separation from nearest data.

$$F(y) = \frac{1}{1-e^{-y}} \tag{1}$$

The Logistic regression is a supervised machine learning algorithm used to determine the probability of a target variable. In the logistic regression the sigmoid function is used. There are three logistic regression types: the first type is binomial or binary, in this type a dependent variable has two possible values 0 or 1 such as yes or no, success or failure, etc. Multinomial is the second type which has more than two unordered values, such as type A or B or C. the final type is the ordinal that has more than two ordered values such as good, very good, excellent and the score of each category like 1,2,3. The logistic function is the core of the logistic regression technique and is defined as in Eq. 1:

Where $y = w_0 + w_1.x_1 + w_2.x_2 + \ldots + w_n.x_n$. The regression coefficients $w_0, w_1, w_2, \ldots w_n$ are calculated using Maximum Likelihood Estimation, and $x_1, x_2, x_3, \ldots, x_n$ represent the features. [43]. The following techniques use logistic regression for classification [38, 44–46].

The K-nearest neighbors (KNN) is a supervised machine learning technique used for solving regression and classification problems. The following techniques use KNN for classification [34, 47, 48, 40], the Pseudo Code of KNN is shown below. [49]

KNN Pseudo Code

1. Load the data.
2. Initialize the value of K.
3. For each sample in the training data.

    1.1. Calculate the distance between each row of training data and test data.

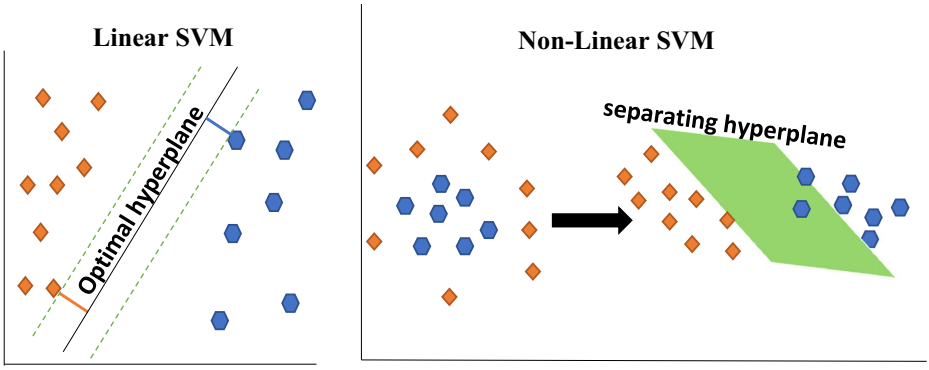

Fig. 4 Illustration of SVM algorithm

1.2. Sort the distances in ascending order.
1.3. Pick the top k rows.
1.4. Pick the most repeated class of these rows.
1.5. Return the class label.

Although the traditional classification methods have been used in many problems, the classification accuracy is unsatisfactory. While the deep learning models have a powerful ability in learning that integrates the process of feature extraction and classification into one model which improves the classification accuracy. The most popular deep neural network model for the problem of image classification is Convolutional Neural Networks (CNNs) [50]. The following techniques use CNN for classification [51–54].

1.1 The list of contributions

This paper presents the following contributions:

- A comprehensive review for salient object detection models has been presented. This paper includes many methods that have not mentioned in previous work such as weakly-supervised methods, omni supervised methods, transformer-based methods.
- In addition to semantic and instance segmentation, panoptic segmentation, and the recent panoramic panoptic segmentation have been presented.
- Visual saliency detection in adverse conditions such as the nighttime are presented in this paper.
- Popular datasets for salient object detection have been discussed in detail.
- F-measure, accuracy, and S-measure between 57 approaches has been presented. Also, a computational complexity comparison in terms of elapsed time for each method is presented.
- Comparative study in terms of model, dataset, execution time and result has been introduced.

The rest of the paper is organized as follows; in section 2 the popular datasets for SOD are presented. In section 3, evaluation matrices are described. In section 4, the feature-based approaches are discussed. Section 5 contains a review of deep learning-based models. Section 6 includes the performance evaluation for 41 models. And the conclusion is finally presented in section 7.

2 Datasets

There are a lot of images datasets used for SOD for example DUTS which is a large-scale dataset that includes challenging scenarios for salient object detection collected from ImageNet DET datasets that contains mammals and vehicles images. It includes 15572 images divided into 10,553 images for training and 5,019 images for testing. Another example is SOC dataset which includes images with salient objects from daily life object categories in real-world scenes. It provides 6000 images (3600 for training, 1200 for validation, and 1200 for testing). Also HKU-IS dataset consists of 4447 challenging images with pixel annotations of salient objects most of these objects have either multiple salient objects or low contrast. The

PASCAL-S dataset consists of 850 images which is used to evaluate the Models performance over images with cluttered backgrounds and multiple objects on the scene. Complex scene saliency dataset (CSSD) [55] contains only 200 images and the extension of CSSD is ECSSD dataset which includes 1000 images with complex scenes collected from the internet. Finally, DUT-Omron dataset consists of 5,168 images with high quality. These images have one or more salient objects and complex backgrounds.

There are also a lot of video datasets used for SOD Such as FBMS-59 which includes 59 videos for motion segmentation, DAVIS-2016 includes 50 videos for video segmentation with pixel-wise annotations per-frame, and VOS includes 200 indoor/outdoor videos that are divided into VOS-E which contains 97 easy videos with obvious foreground objects and VOS-N which contains 103 normal videos with complex foreground objects. Table 1 Illustrates the details of the SOD datasets.

3 Evaluation metrics

The evaluation of the SOD Models is based on Mean Absolute Error (MAE), F-MEASURE (F1 score), and Structure-measure (S-MEASURE), the mathematical background behind each of them is described in detail.

Mean absolute error (MAE) MAE is a loss function often used in evaluating vector-to-vector (also known as multivariate) regression models. The loss function is defined in Eq. 2.

$$\mathrm{LMAE}\left(\mathrm{A}, \mathrm{A}^{*}\right) = \frac{1}{N}\sum_{i=0}^{N}\|x_i - y_i\|1 \tag{2}$$

Where N is the length of predicted vectors A= $\{x_1, x_2, x_3 \ldots, x_N\}$ and $A^* = \{y_1, y_2, y_3 \ldots, y_N\}$. [64]

F-MEASURE F-measure is also called F1 score which is used for assessing the performance of algorithms especially when dataset is imbalanced; it is a harmonic mean of precision and recall. The confusion matrix for an unbalanced dataset is shown in Table 2. [65]

Precision and Recall are required for calculating F-measure and their equations are as in Eq. 3, Eq. 4:

Table 1 Overview of the SOD image and video datasets

#	Dataset	Research	Year	Images	Annotation	Image/Video
1	DUTS [15]	[26, 56]	2019	15572	Pixel-wise	I
2	SOC [19]	[56, 57]	2019, 2020	6000	Pixel-wise	I
3	HKU-IS [21]	[26, 57]	2019, 2020	4447	Pixel-wise	I
4	PASCAL-S [23]	[25, 27]	2019	850	Pixel-wise	I
5	ECSSD [22]	[25, 27]	2019	1000	Pixel-wise	I
6	DUT-Omron [20]	[26, 25]	2019	5,168	Pixel-wise	I
7	FBMS-59 [58]	[59, 60]	2019	59 videos 720 frames	Pixel-wise	V
8	DAVIS-2016 [61]	[62]	2018	50 videos 3455 frames	Pixel-wise	V
9	VOS [63]	[60]	2019	200 videos 116,103 frames	Pixel-wise	V

Table 2 The confusion matrix

		Predicted class	
		Yes	No
Actual Class	Yes	True Positive (TP)	False negative (FN)
	No	False Positive (FP)	True negative (TN)

$$\text{Precision} = \frac{TP}{(TP + FP)} \tag{3}$$

$$\text{recall} = \frac{TP}{(TP + FN)} \tag{4}$$

The F-measure calculated as in Eq. 5:

$$\text{F}-\text{measure} = \frac{2 \times precision \times recall}{precision + recall} \tag{5}$$

S-MEASURE Structure-measure simultaneously assesses object-aware and region-aware similarity between ground-truth (GT) map and saliency map (SM). The formulation of s-measure is described in Eq. 6. [66]

$$\text{S}-\text{measure} = \frac{2a\,b}{(a)^2 + \left(b\right)^2} \cdot \frac{2\sigma_a\sigma_b}{{\sigma_a}^2 + {\sigma_b}^2} \cdot \frac{\sigma_{ab}}{\sigma_a\sigma_b} \tag{6}$$

Where a= $\{a_1, a_{2....} a_N\}$ and b= $\{b_1, b_{2....} b_N\}$ be the pixel values of SM and GT, respectively. a,b, σ_a, σ_b are the mean and standard deviations of a and b. σ_{ab} is the covariance between a and b.

4 Feature-based approaches for SOD

The models were grouped into three subgroups block-based models with intrinsic cues, area-based models with intrinsic cues, and models with extrinsic cues (area and block based). Feature descriptors are also presented [3].

4.1 Block-based methods with intrinsic cues

In this subsection, salient object detection methods which use intrinsic cues from blocks is reviewed. Salient object detection is defined as detecting the rarity, uniqueness, or distinctiveness in a scene [67]. The rarity was calculated as the pixel-based center-surround contrast. The approaches which based on patches or pixels have two shortcomings: i) the edges of the high contrast usually stand out and this prevents the detection of salient object. ii) When using blocks of large size, the salient object boundary is not preserved well [3]. Therefore, the need

for different methods to overcome these problems appeared and the region- based methods will be discussed in the next subsection.

4.2 Region-based models with intrinsic cues

The region-based segmentation techniques are a segmentation technology depends on discovering regions directly. There are two main types of region-based extraction techniques: the first is region growth, which begins from one pixel then gradually combines to form the required segmentation region; the second one is split and merge (Quadtree Method) whose task is to cut the required segmentation region from the overall situation step by step [68, 69].

The region based models for salient object detections adopt signals from image regions created using techniques such as mean-shift [70, 71], TurboPixels [72], graph-based segmentation [73–75], or Simple Linear Iterative Clustering (SLIC) [76, 77].

The regional saliency score is the average score of its segmented patch and contained pixels based on multi-scale contrast [78]. Gargi Srivastava and Rajeev Srivastava [79] compute the score of background for each region based on the difference between the image regions and the feature vector of the boundary.

4.2.1 Mean Shift algorithm

The mean shift is a non-parametric clustering algorithm with no assumptions and used in unsupervised learning. The data points are assigned to the clusters iteratively and the points are shifted towards the highest intensity of data points. It needs only one parameter called bandwidth which determines the clusters number automatically. The multivariate kernel K(a) is calculated as in Eq. 7. [80]

$$K(a) = \left\{ \frac{n}{h_f{}^{f_d} h_s{}^{s_d}} \left(\left| \left| \frac{a^s}{h_s} \right| \right) K_X \left(\left| \left| \frac{a^f}{h_f} \right| \right) \right) \right\} \tag{7}$$

In the above equation, f_d and s_d represent the feature dimensionality and spatial domains, respectively, and h_f and h_s are the kernel bandwidths and are set by the features vector, n is the constant of normalization and K_X is the Epanechnikov-kernel which is shown in Eq. 8.

$$K_X(a) = \begin{cases} 1-a, & \|a\| \leq 1 \\ 0 & otherwise \end{cases} \tag{8}$$

In [81–84] the mean shift algorithm is used. Fatemi, et al. [81] use the color space in the feature extraction step. Based on features, the clusters are created. Then, the mean shift algorithm is used to detect and separate the maximum pixels of the object from the background. Xia, et al. [82] Use a regional based algorithm for salient object detection. First, the image segmentation is done using a superpixel algorithm and for each region, the feature vector is extracted. A mean shift algorithm of ten different bandwidth is used to obtain the Clustered region and ten clustered maps. After that, ten saliency maps are generated from the calculation of the ten clustered maps. XGBoost model is used to merge the ten saliency maps into one saliency map. Fatemi, et al. [83] use a mean shift algorithm for object detection. The steps can be as follows: the image segmentation is applied according to their segment border distance, segment distance and color, five features are obtained. Next, the classification is performed by the

mean shift algorithm according to the similarity of their feature and the output image is created for the object detection by setting a salient score for each cluster. Shen and Wu [84] used mean shift algorithm for image segmentation. For each segment, the average saliency is computed. Also, for the entire image the total mean saliency value is obtained. If the average saliency in the segment is greater than twice of the total mean saliency value, this segment is considered as foreground.

4.2.2 Simple linear iterative clustering (SLIC)

The SLIC algorithm is used for clustering pixels and generates super pixels based on the proximity in the plane of the image and the color similarity. This is applied in the five-dimensional space. So, the spatial distances must be normalized to use the Euclidean distance in the five-dimensional space. [85]

In 2015, Tong, et al. [86] used the SLIC algorithm to generate the super pixels and construct the weak saliency map. Then, SVM classifier is used to detect salient pixels.

4.2.3 Multi-scale contrast

$$f_c(y, I) = \sum_{m=1}^{M} \sum_{y' \in W(y)} \left\| I^m(y) - I^m\left(y'\right) \right\|^2 \tag{9}$$

Contrast is the most widely used local feature for salient detection because the contrast factor mimics the human optical receptive fields. It is usually calculated at multiple scales without knowing the salient object size. The feature of multiscale contrast is defined as in Eq. 9. [87]

Where I^m is the mth-level image in the Gaussian image pyramid and M represents the number of pyramid levels. W(y) represents the window.

4.2.4 Frequency and spatial domain analysis

Frequency analysis offers an opportunity to manipulate the global information in the input image. This analysis is based only on the Fourier Transform. While the spatial domain deals with salient pixels and local information in the image. Li, et al. [88] proposed a new model for saliency detection by merging global information from frequency domain and local information from spatial domain. The non-salient regions are modeled in the frequency domain analysis instead of modeling salient regions. While in the spatial domain those informative regions are enhanced through using a center-surround mechanism. Finally, the saliency map is produced by combining the outputs from these two analysis channels.

4.2.5 Window composition

The window composition is one of salient object detection techniques. The image is segmented, and window composition is measured. The saliency score of a window is used to determine how likely the window includes a salient object. The score function is applied to all windows of different object sizes in the whole image and windows with maximum score are detected as salient objects. This detector is more likely to achieve high results than utilizing an

intermediate saliency map as it works on the whole original image and searches the window space. [89]

4.3 Models with extrinsic cues

In the third subgroup, models that adopt external cues will be discussed. In addition to the visible signals observed from one input image, the external signals may be derived from the ground truth captions of the training images, video sequence, similar images, or a group of input images of common salient objects [3].

4.3.1 The detection of salient objects with similar images

With a growing large amount of visible content which is available on the web, the detection of salient object has been studied in recent years with images which are visually like an input image. In general, given the image O, S similar images $C_O = \{O_s\}S_{s=1}$ are retrieved firstly from a large set of images C. The salient object detection on the input o can be helped by checking up these similar images.

4.3.2 Co-saliency object detection

Co-salient object detection (CoSOD) is a recently emerging and growing branch of SOD. Instead of focusing on computing the saliency on only one image, the algorithms of CoSOD concentrate on detecting the salient objects which is shared by various input images $\{X^m\}Y_{m=1}$. This means that such objects may be of the same category which shares similar visual appearances or the same object with various viewpoints. The main feature of co-salient detection algorithms is that their inputs are a collection of images, whereas traditional models of salient object detection need only one input image [89, 90].In 2020, Fan, et al. [91] build a new dataset called CoSOD3k, which is a high-quality dataset consisting of co-salient objects, the similarities are in the level of conceptual or semantic. What is worth mentioning, CoSOD3k is the most challenging Co-salient object detection dataset so far that consists of 160 groups and 3,316 images annotated with classes, instance-level annotations, object-level, and bounding boxes. Moreover, a comprehensive study is also provided by summarizing 34 algorithms of cutting-edge algorithms, 19 of them are benchmarked over the CoSOD3k dataset in addition to other 4 existing datasets.

4.4 Feature descriptor

The development of feature descriptors has received attention and attracted a lot of researchers. The main idea of feature descriptor algorithm is that it takes an image as an input then outputs feature vectors or feature descriptors [92]. The feature descriptors that are commonly used are Histogram of Orientated Gradients (HOG) [93] , Scale Invariant Feature Transform (SIFT) [94, 95], Speeded-Up Robust Features (SURF) [96], Gray-Level Co-occurrence Matrix (GLCM) [97, 98], and Local Binary Patterns (LBP) [99, 100]. A review of each feature descriptor algorithm will be provided.

4.4.1 Histogram of orientated gradients (HOG)

HOG descriptor is a well-known appearance and shape algorithm. It was invented in 2005 by Dalal and Triggs to reveal a pedestrian in an image. The idea of the Hog algorithm is to locally compute the gradient orientation for each pixel and extract the normalized histogram features over overlapping and dense grid. [101] The steps of HOG algorithm is illustrated in Fig. 5. In the HOG algorithm, the image is divided into blocks and each block of them is divided into cells. For each cell, the gradient is computed by calculating the horizontal and vertical derivatives using the Surrounding pixels. The gradient is computed as illustrated in Eq. 10 and Eq. 11.

$$G_x(x,y) = L(x+1,y) - L(x-1,y) \tag{10}$$

$$G_y(x,y) = L(x,y+1) - L(x,y-1) \tag{11}$$

Where L (x, y) is the image pixel at the location x and y, G_x is the horizontal gradient and G_y is the vertical gradient. The second phase in the HOG algorithm is to compute the orientation and magnitude of each pixel as illustrated in Eq. 12 and Eq. 13.

$$\text{Magnitude } (x, y) = \sqrt{{G_x}^2 + {G_y}^2} \tag{12}$$

$$\text{Orientation } (x, y) = tan^{-1}\frac{G_y}{G_x} \tag{13}$$

The third step is to create the histogram which is based on the calculated pixel's orientation in each cell that could be in range of 0 to 180 or 0 to 360 depending on the configuration of the implementation. A histogram is a plot used to illustrate the frequency distribution of the continuous data. On the x-axis, the orientation or angle in the form of bins is exist and the frequency is on the y-axis.

In the fourth step, the histogram normalization for each cell within each block is applied. Finally, concatenating all values of the histogram in the selected window to obtain the Hog features. [102]

4.4.2 Scale-invariant feature transform (SIFT)

The SIFT is an algorithm to detect and identify the local features in images. It was invented in 1999 by David Lowe. SIFT can transform images into feature vectors which are used for recognizing objects. Five stages are applied for SIFT descriptor: [103]

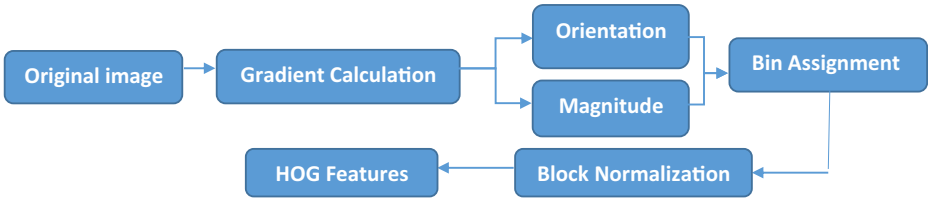

Fig. 5 Steps of Hog Algorithm

Detection of scale-space extreme In the first step, several octaves of the data image are generated. Gradually blurred out images are generated from the original image. After that, the original image is resized to half size. Then blurred out images are generated again. And this step is repeated. The scale range is computed based on Eq. 14.

$$B\,(\mathrm{a},\mathrm{b},\sigma) = G\,(\mathrm{a},\mathrm{b},\sigma)*I\,(\mathrm{a},\mathrm{b}) \tag{14}$$

Where B is a blurred image, G is the operator of Gaussian Blur, I is the data image, a, b are the location coordinates, σ is the amount of blur or scale parameter. The greater the value is, the higher the blur is. And the * is used to apply Gaussian blur G at location a, b of the image I. [104]

LoG approximations In the second step of SIFT, Difference-of-Gaussians (DOG) which is a Laplacian-of-Gaussian (LoG) estimate, is applied to identify keypoint invariants and location in the scale space. Localizing scale space D (a, b, σ) by computing the variance in two images, one of them with scale h multiplies the other. Eq. 15 illustrates the difference between two Gaussians. [105]

$$D\,(\mathrm{a},\mathrm{b},\sigma) = G\,(\mathrm{a},\mathrm{b},\mathrm{h}\sigma) - G\,(\mathrm{a},\mathrm{b},\sigma) \tag{15}$$

Keypoint localization & filtering In the third step, the minimum and maximum values of DOG images is detected with comparing its eight neighbors in the current scale, and the 9 neighbors in the scale below and the scale above. If this value is the greatest or lowest value of all other points, so this point is an extremum. After that, the keypoints number is reduced by removing keypoints of low contrast or keypoints that located on the edge. This done by computing the Laplacian as illustrated in Eq. 16 [105]

$$Z = -\frac{\partial^2 D^{-1}}{\partial X^2}\frac{\partial D}{\partial X} \tag{16}$$

Keypoint orientations Keypoint orientations is the fourth step. The goal of orientation assignment is to designate a specific orientation to the keypoints based on the characteristics of the local image. To apply orientation assignment the histogram and small area around it is used. After creating an orientation histogram, the most salient gradient orientation(s) are detected. If there is just one peak, it is designated to the keypoint. But in case of several peaks above 80% mark, all of them are transformed into a new keypoint taking into consideration their respective orientations. [106]

Generating a feature vector This is the final step of SIFT. So far, the rotation and scale invariance are calculated. The remaining is to differentiate between keypoints by making a fingerprint for every keypoint. A 16x16 window of pixels around the keypoint is split into sixteen 4x4 windows. A histogram of 8 bins is generated from each 4x4 window. The first bin contains gradient orientations in the range 0 to 44 degrees. The second bin contains gradient orientations in the range 45 to 89 degrees and so on. This is done for all 16 pixels (4x4 blocks) and sixteen 4x4 regions. So, you finally need 4x4x8 = 128 numbers. Once 128 numbers are gotten, normalize them to get the feature vector ready. The final feature vector consists of the 128 normalized values. [106]

4.4.3 Speeded-up robust features (SURF)

Speeded up Robust Features (SURF) is one of the computer vision techniques which is used for classification and object recognition. SURF follows the idea of SIFT algorithm, But SURF is faster and robust when it is compared to SIFT. In SIFT, approximated LoG with DoG for detecting scale-space. But SURF approximates LOG with use of Box filter. SURF depends on Hessian matrix determinant for location and scale. The hessian matrix is used to find the max value. For each point A= (a, b) in image I, the Hessian Matrix H (A, σ) inside A, on the σ scale determined as the formula in Fig. 6.

$$\mathrm{H}\ (\mathrm{A}, \sigma) = \begin{vmatrix} L_{aa}(\mathrm{A}, \sigma) & L_{ab}(\mathrm{A}, \sigma) \\ L_{ab}(\mathrm{A}, \sigma) & L_{bb}(\mathrm{A}, \sigma) \end{vmatrix}$$

Where L_{aa}(A, σ)is the second Gaussian convolution from image I at point A and the same for L_{ab}(A, σ) andL_{bb}(A, σ). Wavelet responses is used for feature description task in SURF for both vertical and horizontal direction as shown in Eq. 17.

$$\mathrm{V} = (\sum d_a,\ \sum d_b, \sum |d_a|,\ \sum |d_b|). \tag{17}$$

When this equation is represented in a form of vector, the SURF feature descriptor is obtained [107].

The performance evaluation and elapsed time of different feature-based approaches for SOD in different SOD datasets is shown in Table 3.

4.4.4 Local Binary Pattern (LBP)

$$\begin{aligned} \text{LBP code} &= \textstyle\sum_{(over\ n)} F(I_n - I_{thresh}) \times 2^n, \\ \mathrm{F(x)} &= \begin{cases} 1 & if\ x \geq 0 \\ 0 & if\ x < 0 \end{cases} \end{aligned} \tag{18}$$

Local Binary Pattern (LBP) is a simple texture operator that describes the relation between pixel and its neighbors. The most common LBP approaches is that each 3×3 window is processed to extract the code of LBP. the processing includes thresholding the pixel in the center of window with its around pixels using the window median, window mean, or the center pixel itself as thresholds. Then, the code of LBP is given by Eq. 18. [108]

Where I_thresh is the threshold value and I_n are the surrounding window pixels intensities with (n=0, 1, 2…,7). The algorithm steps of LBP as follow:

Local Binary Pattern steps

1- Convert the image to grayscale image.
2- For each pixel in the image, select a neighbor of size r surrounding the center pixel.
3- For each pixel, compare the center value and the neighbor. In binary operation, if the neighbor values are less than the center, record 0 else record 1.
4- Convert the binary value to decimal.

Fig. 6 Hessian Matrix

$$\mathrm{H}\ (\mathrm{A}, \sigma) = \begin{vmatrix} L_{aa}(\mathrm{A}, \sigma) & L_{ab}(\mathrm{A}, \sigma) \\ L_{ab}(\mathrm{A}, \sigma) & L_{bb}(\mathrm{A}, \sigma) \end{vmatrix}$$

Table 3 Performance evaluation and elapsed time of feature-based approaches

#	Model	Year	Dataset	Accuracy	Time per frame (s)
1	ORB [111]	2015	LFPW	75.08%	0.3962
2	ORB [112]	2017	Visual_Indoor	63.62%	38.04
3	ORB-SURF [112]	2017	Visual_Indoor	97.90%	141.77
4	ORB-BRIEF [112]	2017	Visual_Indoor	62.83%	38.04
5	HOG [113]	2013	real-time video	97.4%	0.051
6	HOG [114]	2020	mini-MIAS	99.86%	0.16
7	HOG [115]	2019	PASCAL VOC	46.0%	23.09
8	SIFT [113]	2017	Visual_Indoor	62.31%	81.99
9	SIFT_SURF [113]	2017	Visual_Indoor	98.41%	63.06
10	SIFT [111]	2015	LFPW	69.72%	0.9276
11	SIFT [116]	2016	Zurich building	69.7%	-
12	SIFT [117]	2020	PlantVillage	87.25%	2.062
13	SIFT [116]	2016	Kentucky	48.2%	-
14	SURF [112]	2017	Visual_Indoor	89.54%	141.31
15	SURF-BRIEF [112]	2017	Visual_Indoor	62.36%	57.638
16	SURF [111]	2015	LFPW	65.27%	0.6854
17	SURF [117]	2020	PlantVillage	85.73%	1.762
18	SURF [118]	2016	Caltech-101	70.5%	-
19	BRIEF [116]	2016	Zurich building	70.5%	-
20	BRIEF [116]	2016	Kentucky	41.6%	-

4.4.5 Binary robust independent elementary features (BRIEF)

Binary Robust Independent Elementary Features (BRIEF) is a fast feature descriptor algorithm that cannot identify the key points by itself, so it is combined with key point detector. it uses a smoothed image; pixel pairs are selected and the gray values between them are compared. Test (t) on patch (P) of size (M×M) is defined as in Eq. 19.

$$\mathrm{T}\ (\mathrm{P};\mathrm{a},\mathrm{b}) := \begin{cases} 1 & if I(p,a) < I(p,b) \\ 0 & otherwise \end{cases} \tag{19}$$

Where I (p, a) is the intensity of the pixel in a smoothed version of p. Define a set of binary tests by choosing a set of $n_d(a,\ b)$ -location pairs. The BRIEF descriptor is taken to be the n_d dimensional bit that corresponds to the decimal representation in Eq. 20. [109]

$$\sum_{1 \le i \le n_d} 2^{i-1} T(p; a_i, b_i) \tag{20}$$

4.4.6 Oriented FAST and rotated BRIEF (ORB)

ORB is a very robust binary descriptor which based on FAST feature detector and BRIEF binary descriptor. The algorithm steps are as follows: [110]

ORB algorithm
1. Reduce the size of the input image to various scale levels.
2. On all levels, extract FAST features.
3. Apply grid filtering.
4. Feature orientation extraction.
5. Descriptor's extraction.

4.5 Saliency detection in adverse conditions

While saliency detection methods achieve high accuracy rate in many complex problems, research shows that there may be unexpected situations or adverse conditions such as saliency detection in the nighttime. In [119], a novel Bayes saliency detection method was proposed for nighttime RGB traffic images. In this approach, prior estimation, weight estimation, feature extraction, and Bayes rule are applied to compute saliency maps. Then, an effective and simple object proposal generator which is based on the Bayes saliency map was proposed. In [120], the gap between day and nighttime images for semantic segmentation was analyzed. Generative Adversarial Networks (GANs) method was used to propose two methods to tackle the gab. On the other hand, during night inference, night to day conversion was performed to convert the input data into a suitable domain that were trained in daylight images.

5 Deep learning models for SOD

Convolutional Neural Networks is the most popular deep learning model for image recognition and classification. The components of CNN and different CNN architectures are presented below. [32, 33]

5.1 Convolutional neural networks (CNNs)

CNN is one of the most used tools in the field of machine learning. Many vision problems are solved using CNN such as semantic segmentation [121, 122], edge detection [123], and object recognition [124]. Also, most recent research has shown that CNNs are effective in SOD [125].

5.1.1 CNN Components

The CNN consists of several types of layers:

Convolutional layer The convolutional layer consists of set of neurons where each neuron represents a kernel. The kernel works by splitting the image into blocks which helps in feature extraction. Kernel used a set of weights to convolve with the images by calculating a dot product between the small region and the weights. The operation is shown in Eq. 21.

$$f_l^m\,(\mathrm{a},\mathrm{b}) = \sum_h \sum_{x,y} i_h\,(\mathrm{x},\mathrm{y}).e_l^m(\mathrm{t},\mathrm{v}) \tag{21}$$

Where f_l^m (a, b) represents a (a, b) element of feature matrix for lth layer and mth neuron, i_h (x, y) represents (x, y) element of hth channel of an image i and e_l^m(t, v) represents (u, v) element of mth kernel of lth layer.

Pooling layer (downsampling) In this layer, the amount of information which was generated by the convolutional layer is scaled down and the most important information is maintained.

Activation function It is a decision function which helps in complex pattern learning. The appropriate activation function selection can accelerate the process of learning. There are

several activation functions such as tanh, sigmoid, SWISH, maxout, ReLU, and different types of ReLU such as leaky ReLU and ELU.

Batch normalization The normalization is applied to address the problem related to the distribution change of hidden units' values. After applying the normalization, the distribution of feature map values is unified by setting them to 0 mean and unit contrast batch normalization.

Fully connected layer (FC) FC layer is the final layer in the network used for the data classification and the determination of the class label. [126]

In 2020, Üreten, et al. [127] developed a CNN method to help doctors during the diagnosis of rheumatoid arthritis. The dataset which contains 180 radiograph images from the faculty of medical in Kırıkkale University is used in this study. 81 patients of them are normal and the 99 patients suffer from RA. The dataset is divided into 135 radiographs images for training (74 RA and 61 normal) and 45 for testing (25 RA and 20 normal). The results show that the network achieved 73.33% accuracy, 0.0167 error rate, 0.6818 sensitivity, 0.7826 specificity and 0.7500 precision.

5.1.2 CNN Architecture

Over the years, CNN have evolved greatly, and the most important CNN models will be presented. [128]

LeNet-5 LeNet-5was created in 1998 by LeCun [129]. It consists of two convolutional layers, three fully connected layers and has a subsampling layer which is currently known as pooling layer. It contains around 60,000 parameters and is widely used for handwritten digits recognition.

The first layer is the input layer which is built to take 32x32, and these are the input dimensions to the next layer. The grayscale images used in [129] had normalized their pixel values from (0 to 255), to values between (-0.1 and 1.175) to ensure that the images batch have standard deviation of 1 and mean of 0 to reduce the training time. The first convolutional layer C1 outputs 6 feature maps with dimensions 28x28, and the size of the kernel is 5x5. The layer that follows C1 is the subsampling layer S2 and known as downsampling layer. This layer halves the dimension of feature maps which were received from the previous convolution layer and outputs 6 feature maps of size 14×14. The third convolution layer is C3 with 16 convolution kernel of size 5×5. The input of the first six feature maps of C3 is continuous subset of the three S2 feature maps, the input of the following six feature maps is the four continuous subsets input, and the input of the input of the following three feature maps is the four discontinuous subsets. The last feature map input is all S2 feature maps. Layer S4 and S2 are similar but S4 with size of 2 ×2 and 16 feature maps with dimensions 5×5. The convolution layer C5 is with 120 convolution kernels with kernel size equal 5×5. The fully connected layer F6 is connected to C5, and the outputs are 84 feature maps.

AlexNet AlexNetwas developed in 2012 by Krizhevsky, Hinton, and Sutskever [130]. The AlexNet architecture won the 2012 ImageNet competition and consists of three fully connected layers and five convolutional layers. The architecture of AlexNet is like LeNet-5

architecture, only much deeper and larger, and rather than stacking a downsampling layer on top of every convolutional layer, it was directly stack convolutional layers on top of each other. AlexNet contains around 60 million parameters. In AlexNet, Rectified Linear Units (ReLUs) is used as activation functions.

The first layer is a convolution layer, the size of the input image is 224×224×3, and number of filters is 96 with filter size 11×11×3. The second layer is a max pooling layer with 55×55×96 input and 256 filters of size 5×5×48. The network adds max pooling layers with window of 3×3 and a strid of 2 after the first, second, and fifth convolution layers and layers 3,4,5 follow on the same lines. Layer 6 is a fully connected layer with input 13×13×128 and output 2048. Layers 7, 8 follow on the same lines. There are two Fully connected (FC) layers after the last convolution layer. These huge FC layers include 4096 outputs.

VGG-16 In 2014, Visual Geometry Group (VGG) [131] introduced the VGG-16 which consists of 13 convolutional layers and 4 fully connected layers. Just like AlexNet, VGG-16 used ReLU as activation function. It contains 138 million parameters. Developers of VGG also developed a deeper version called VGG-19.

The input to VGG is RGB image of 224 x 224. in the first two layers, there are 64 channels of 3×3 filter dimensions and stride 1. After max pooling layer of stride 2, there are a stack of convolutional layers with different depth in different architectures. Layers 3 and 4 are convolution layers using 128 filters. After max pooling, there are three convolution layers with 256 filters. After max pooling, there are six convolution layers with 512 filters with max pooling after each three layers. Then, there are three fully connected layers: each of the first two layers have 4096 channels, and the third contains 1000 channels. The SoftMax layer is the last layer.

ResNet-50 The residual network (ResNet-50) was the winner of the 2015 ILSVC challenge and created by He, et al. [132]. It contains 26 million parameters and 50 layers, each ResNet block have two or three convolutional layers.

The input image to the network has height and width as multiples of 32. In the first layer of ResNet architecture, the convolution using 7×7 kernel sizes and 64 different kernels, followed by max-pooling using 3×3 kernel sizes and stride size of 2. Afterward, there are 3 Residual blocks with 3 layers each. The kernel size in all 3 layers is 64, 64 and 128 respectively and the convolution operation in each Residual block is applied wit stride 2. Finally, there are average pooling layers followed by fully connected layers that have 1000 nodes and softmax function at the end.

Basically, deep-learning based SOD models can be divided into two categories: the first category is Classic Convolutional Network (CCN-based) models. The second one is Fully Convolutional Networks based (FCN-based). In CCN-based models, the multi-layer perceptron (MLP) is used for the detection of saliency. The input image is over-segmented into small regions. Then, the high-level features are extracted using CNN. After that, these features are fed to a MLP to determine the value of saliency of small region. Unlike FCNs, the spatial information from features that are extracted from CNNs cannot be preserved due to the use of MLPs.

5.2 Classic convolutional network (CCN)-based models

CCN models is the two-stage object detection models such as Region-based Convolutional Networks(R-CNN) [133], Fast R-CNN [134], Faster R-CNN [135], and Mask R-CNN. [136]

The R-CNN family contains several models for object detection. All these models are region-based and have achieved significant development with time, resulting in increased efficiency and accuracy. [137]

5.2.1 Region-based convolutional networks (R-CNN)

The R-CNN model is based on selective search to detect the objects in the image. This model divides the image into huge number of regions and working collectively on them. These collections are checked if they have any object or not. The main factor for the success of this method is the accuracy of object classification.

In 2020, Ucar, et al. [138] proposed a Regions-based Convolutional Neural Network (R-CNN) model for the detections of aircraft using the satellite images from airports in Turkey. The dataset contains 32,000 images: 8,000 images from the class plane and 24,000 images from the class not aircraft (NPlane). For balance, only 16,000 images are used from equal number of plane and NPlane. The results show that the model achieved %98.40 accuracy, %98.62 sensitivity, %98.13 Specificity, 0.0187 FP Rate, and 0.0138 FN Rate.

5.2.2 Fast R-CNN

The Fast R-CNN model uses the R-CNN structure with the Spatial Pyramid Pooling (SPP-net) to make the R-CNN faster. The SPP-net is used in Fast R-CNN to calculate the representation of CNN only once for the whole image. This representation is then used to calculate the representation of CNN for each region generated by the approach of selective search. The bounding box regression is also included in the Fast R-CNN Model along with the process of training. This makes both of classification and localization processes in a single process which make the process faster.

5.2.3 Faster R-CNN

The Faster R-CNN model is faster than the Fast R-CNN. The selective search process in Fast R-CNN is replaced with Region Proposal Network (RPN) in Faster-RCNN. The RPN implements a small convolutional network which makes the selection process faster and generates area of interest. Anchor Boxes is also used in this method along with RPN to handle the scale of objects and multiple aspect ratios. Faster R-CNN is considered one of the most efficient and accurate object detection models.

In 2020, Wei, et al. [139] proposed a detection and classification model for breast cancer with use of VGG-16 and Faster Region-based CNN (Faster R-CNN) on OASBUD dataset of ultrasonic images. This dataset contains 200 ultrasonic images for both training and testing. There are 112 ultrasound images for training, 48 images for validation and 40 images for testing with no duplicate images; each ultrasound image has only one region of interest (ROI). The results show that Faster R-CNN improve the performance and obtain accuracy more than 95%.

In 2020, Alalharith, et al. [140] developed a deep learning model for the periodontal disease detection in orthodontic patients. Two faster Region based CNN using ResNet-50 network were developed; one model for the teeth detection to determine the region of interest (ROI) while the other model for the gingival inflammation detection. The dataset was obtained on 7 October 2019 from the Dentistry College, university of Imam Abdulrahman bin Faisal. It

contains 134 intraoral images which divided into 107 images for training and 27 imaged for testing. The results show that the model of teeth detection achieved 100% accuracy, 100% precision, 51.85% recall, and 100% mAP, while the model of the inflammation detection achieved 77.12% accuracy, 88.02% precision, 41.75% recall, and 68.19% mAP.

In 2020, Li, et al. [141] proposed a deep learning architecture for rice disease detection in videos. The video is first transformed into still frame, then the frame is sent to a detection model based on faster region-based CNN. In this study, the dataset was collected between June and August 2018 in Anhui, Hunan Province, and Jiangxi, China. It consists of videos and images of rice disease; 1760 images for stem borer symptoms, 1760 images for brown spot, and 1800 images for sheath blight. The dataset was divided randomly into training and test with the ratio of 9:1. In terms of blight Precision, Borer Precision, and Spot Precision Custom DCNN achieved 90.5, 100.0, and 0, respectively. Custom DCNN was also compared to YOLOv3, the sensitivity (Recall) of custom DCNN was greater than YOLOv3.The Blight Recall, Borer Recall, and Spot Recall was 74.1, 45.5, and 75.5 respectively in custom DCNN. However, YOLOv3 had higher precision than custom DCNN; Blight Precision, Borer Precision, and Spot Precision was 100.0, 100.0, and 100.0 respectively in YOLOv3.

In 2020, Zhang, et al. [142] proposed a model for the detection of four tomato diseases: ToMV, powdery mildew, leaf mold fungus and blight. The model which used in this study is based on improved Faster RCNN, the VGG16 network was replaced by ResNet to obtain a feature map. Then, the K-means technique was used for clustering. The dataset is a laboratory data from AIChallenger. This dataset includes 61 categories, the selected images of four tomato diseases includes 4,178 images. The dataset is randomly divided into 60% training, 10%validation and 30% test. In terms of mAP, the results show that faster RCNN, faster RCNN-mobile, and Faster RCNN-res101 achieved 95.83, 94.67, and 97.18, respectively; While in case of using K-means, faster RCNN, faster RCNN-mobile, and Faster RCNN-res101 achieved 97.01, 97.37, and 98.54, respectively.

In 2019, Tourani, et al. [143] proposed a model for the detection of vehicles in videos. This model is based on faster R-CNN. In this study, two datasets were used. The Stanford University provides a cars dataset which includes 16,185 images of 196 vehicles classes which is divided into 8,144 images for training and 8,041 for testing. And the authors of this study provide a testing dataset which contains 928 images. The sensitivity factor of the system is 0.985 and needs 74 milliseconds for vehicles detection in real condition data.

5.2.4 Mask R-CNN

Mask R-CNN is the extension of Faster R-CNN and aims to solve the problems of instance segmentation in computer vision applications i.e., to separate various objects in a video or an image. A mask branch on every region of interest (ROI) must be included in Mask R-CNN for object prediction Along with the bounding box (BB) and class label branches. The final three outputs are: a bounding box coordinates, object mask, and class label. Mask R-CNN detects objects efficiently in the image and in parallel generates a segmentation mask for each object with high quality.

In 2020, Dai, et.al. [144] Proposed a deep neural network model for the detection of intraprostatic lesions (ILS). The segmentation of prostate gland and IL was carried out using T2 weighted (T2WIs) images. The mask R-CNN model applied the prostate segmentation with Dice similarity coefficient (DCS) of (0.88±0.04), (0.86±0.04), and (0.82±0.05); sensitivity of 0.93, 0.95, and 0.95 (true positive rate); and specificity of 0.98, 0.85, and 0.90 (true negative

rate). However, intraprostatic lesions were segmented with Dice similarity coefficient of (0.64 ±0.11), (0.56±0.15), and (0.46±0.15); sensitivity of (0.57±0.23), (0.50±0.28), and (0.33±0.17); and specificity of (0.980±0.009), (0.969±0.016), and (0.977±0.013).

5.3 Fully convolutional networks (FCN)-based models

FCN models is the one-stage object detection models such as OverFeat [145], You Only Look Once (YOLO) [146], YOLO V2 [147], YOLO V3 [148, 149] and edge guidance network (EGNet) [150].

5.3.1 OverFeat

OverFeat is a pioneer method of integrating the object detection, object localization and classification into one CNN. The idea of OverFeat is to use the multi-scale rapid sliding window on the final pooling layer to extract the patch. The patches will be merged according to the score of classification for each patch. In this way, the problems of multi-size objects and complex shape are solved. Compared to RCNN, OverFeat has advantages in speed, but shortcomings in accuracy.

5.3.2 You only look once (YOLO)

The R-CNN family that we discuss above focuses on the split of an image into parts and then concentrates on the parts that may contain an object, whereas the YOLO family focuses on the whole image and uses bounding boxes, then determines the class label. The YOLO family is a fast and powerful object detector. The first version of YOLO is YOLO v1 or YOLO unified, the reason for this name is due to that this model unifies both object detection and the object classification in one detection network. YOLO predicts a restricted bounding boxes number to achieve real time detection. The main idea of YOLO is to divide the input image into MxM parts and having every cell immediately regress the location of bounding box and the score of confidence if the center of the object falls into that cell. Many bounding box regressors is used in each cell of YOLO because of the different sizes of objects. During training, the bounding box regressor with the highest Intersection over Union (IOU) will be compared with the label of ground-truth, so bounding boxes at the same position will handle various scales over time. Meanwhile, each cell will predict the probabilities of Class C, provided the grid cell incudes an object with high confidence score.

In 2020, Mandal, et al. [151] proposed a You Only Look Once (YOLO) model for monitoring traffic footage. The dataset consists of 18,509 images were collected from traffic camera in RITIS, New York State DOT, Iowa 511, Transportation and Development Department of Louisiana, and Iowa DOT Open Data. For the traffic prediction queues, Mask R-CNN achieved 90.5% while the highest accuracy achieved is 93.7% by YOLO.

5.3.3 YOLOV2

The first YOLO version contains many Shortcomings such as low localization accuracy because of the predictions which is based on coarse grid and the small objects may be difficult to be recognized because the cell contains two scale agnostic regressors. YOLO v2 is also called YOLO9000 and improves the shortcomings of YOLO v1 by focusing on the

localization and recall. The batch normalization, high resolution classifiers, anchor boxes, fine-grained features and multi-level classifiers is used in YOLO v2. A new network called Darknet is used in YOLO v2 for feature extraction, it includes nineteen convolutional layers, five max pooling layers, and finally softmax layer for objects classification. To solve the problem of small objects detection, a pass-through layer is added in YOLO v2 to integrate features from an early layer. It was realized that the resolution of the input is a silver bullet for detecting small object. The input for the backbone not only doubled from 224x224 to 448x448 but also a multi-scale schema of training is invented, that includes various input resolutions at various training periods. YOLO v2 is tested with a version which trained on datasets of 9000 classes hierarchical. Which is an early trial of object detector with multi-label classification. All the mentioned features make YOLO v2 better than YOLO v1.

5.3.4 YOLOV3

YOLO v3 is the final version of YOLO family. It balanced the accuracy, complexity, and speed. Instead of the softmax used by YOLO v2, logistic classifiers are used in YOLO v3 which classifies the objects accurately and makes the multi label classification possible. In YOLO v3 the Darknet53 is used as a feature extractor instead of Darknet19 used by YOLO v2. Darknet53 contains 53 convolutional layers, and this increases the accuracy. V3 also detect small objects in the image which is not possible in YOLO v1. So, YOLO v3 got popular because of all these advantages.

In 2020, Cao, et al. [152] proposed an improved model based on You Only Look Once v3 (YOLO-V3) Algorithm for the detection of the target in the remote sensing images. Ships and airplanes complex backgrounds were selected as MyShip and MyPlane respectively, from DOTA dataset to evaluate the network. In MyPlane dataset, there is 2,368 images; while in MyShip dataset, there is 3,017 images. The results of four network models were compared; in MyPlane, the improved algorithm achieved the best average precision (AP) which is 94.29. While in MyShip, the best AP result is 93.13 for the improved algorithm. The performance of MyShip and MyPlane improved by 1-3% which verifies the effectiveness of the improved algorithm.

In 2019, Zhao and Ren [153] used one-stage object detector called YOLOv3 for the detection of small aircraft of remote sensing images. The dataset consists of 350 remote sensing images which were collected from DOTA dataset and Google Earth. The dataset is divided into 224 images for training, 56 images for verification, and 70 images for testing. The YOLOv3 model was compared with Faster SSD and R-CNN. In terms of AP, the results show that YOLOv3 achieved best result which is 0.925; while in terms of detection time the YOLOv3 achieved 0.023s which mean that YOLOv3 achieved low processing time and excellent accuracy for detection.

5.3.5 Edge guidance network (EGNet)

EGNet is used for detecting salient object in three steps. In the first step, the features of the salient object are extracted by a progressive fusion way. In the next step, the salient edge features are obtained by integrating the information of local edge and global location. Finally, to make use of these features, the salient edge features and salient object features are coupled at various resolutions. [150]

5.4 RGB-D methods

The aim of RGB-D salient object detection (SOD) is to distinguish the most visually distinct regions or objects in a scene from the RGB and depth data. The depth-produced saliency is a helpful complement to color-based saliency models [154]. Attention Complementary Network (ACNet) is one of RGB-D models which selectively collects features from depth and RGB branches. The ACNet consists of three branches based on ResNet; two branches to separately extract features for depth and RGB, several Attention Complementary Modules (ACMs) are used to obtain these features, and the third branch to process the merged features [155]. Another RGB-D method is Complementary Depth Network (CDNet) which consists of four stages: the first stage is RGB and depth encoders, in this stage two independent VGG networks are employed as the encoders for features extraction from RGB and depth image. In the second stage, a new depth map is estimated from the RGB image directly using a decoder to improve the RGB-D SOD performance. In the third stage, the depth feature fusion module selects and merge these depth features from both the original depth and the estimated depth map dynamically. Finally, a two-stage cross-modal feature fusion scheme is applied to take good advantage of depth and RGB features in all levels. [156]

5.5 Instance segmentation

Instance segmentation is the process of detecting and delineating each distinguished object in an image. This problem was raised for the first time in [157], and has been studied a lot in recent years. Inspired by Instance segmentation, salient instance segmentation was proposed in [158] which detects salient regions and recognizes object instances within them. It contains four cascaded steps, including detection of salient region, detection of salient object contour, generation of salient instance and refinement of salient instance. In [159], a framework for saliency detection based on Multiple-Instance Learning (MIL) is presented. In MIL, segmented regions are referred to as bags and the sampled points as instances. Low, mid, and high-level features are incorporated into the process of learning and testing. These features are color, scale, texture, center prior, boundary, and position. after having all these features, a vector is formed for the concatenation of each feature output to train and test the classifier using MIL.

5.6 Panoptic segmentation

The aim of semantic segmentation is to classify each pixel into the appropriate classes, while in instance segmentation, the focus is on segmenting object instances separately. The Panoptic Segmentation (PS) incorporates semantic and instance segmentation to allocate a class label to all pixels and segment all object instances uniquely. In panoptic segmentation, pixel label encoding includes assigning two labels to each pixel in an image: one for semantic label representation and the other for instance id. The pixels with the same label are deemed to be in the same class, and the stuff instance id is ignored. [160]

5.7 Panoramic panoptic segmentations

The most holistic scene understanding, both in terms of field of view and picture level understanding, is panoramic panoptic segmentation. A full understanding of the surrounding provides the agent with the maximum of information, which is crucial for any intelligent

vehicle to make informed decisions in a dynamic and secure environment such as traffic in real-world. The Panoramic Robust Feature (PRF) framework [161] includes two stages: A short pretraining stage that is responsible for providing a robust feature representation to the network's backbone. Following the pretraining stage, standard supervised training on labeled datasets is performed. [161]

5.8 Transformer-based methods

The Transformers is a type of encoder-decoder architecture which uses self-attention layers instead of CNNs or RNNs. the encoder maps a series of inputs into a continuous representation that contains all the input's learned information. the decoder then takes the abstract continuous representation and produces a single output step by step. [162] One of the transformer methods was proposed in [163]. DEtection TRansformer, or DETR, are the key components of the new framework for object detection which is based on bipartite matching loss and transformers. Direct set predictions in detection require two ingredients: (1) a set prediction loss that imposes unique matching between ground truth and expected boxes; (2) an architecture that predicts a set of objects in a single pass and models their relationship. Another vision transformer called Swin Transformer was proposed in [164]. It produces a hierarchical representation which is calculated with shifted windows. The shifted window bases self-attention achieves greater efficiency on computer vision problems. This hierarchical architecture can model at different scales and has linear complexity of computation with respect to image dimensions.

5.9 Context modeling methods

Context is a valuable source of information regarding an object's identity, scale, and location which is important for visual saliency detection. In [165] a method for detecting the boundary of salient objects that takes advantage of depth information was proposed. Because context is vital in saliency recognition, the method combines the multimodal fusion network (MCMFNet) and multiscale multilevel context to aggregate the feature maps of multiscale multilevel context to detect salient objects accurately. Finally, a coarse-to-fine method is performed to an attention module retrieving feature maps of multilevel and multimodal to obtain the final saliency map. In [166], Omnidirectional image segmentation was viewed from the perspective of context awareness. The Efficient Concurrent Attention Networks (ECANets) was proposed to capture the inherent omni-range dependencies which can stretch across 360°. As the horizontal dimension of the wideFoV panoramas include rich contextual dependencies, the ECANet includes a Horizontal Segment Attention (HSA) module and a Pyramidal Space Attention (PSA) module. The model training was upgraded by leveraging multisource omni-supervised learning.

5.10 Selective contrast

Selective contrast explores the most distinctive component information in texture, color, and location. In the human vision system, the opponent colors are present in the early color processing instead of the RGB bands. As a result, a transformation could be applied to get the intrinsic color space component, various linear and nonlinear techniques are available for this task. The transformed color vector expression is called selective color as the representation is more distinctive. In term of texture, different pixels arrangement forms different textures,

that would provide descriptive information. Each texture is referred to as its closest texture prototype, this expression is called selective texture. In [167], it was pointed out that important things are more likely to be in the center of an image. This selection principle is generally implemented by giving more weights to the center area and less weight to the area near the edge. But in [168], the reweighing is applied on regions instead of pixels because calculations that based on region can resist a noise at a certain level. so, the input image is over segmented, and the segmented regions become the key unit for calculating saliency. Then, the center prone prior is added. But it is found that a salient object is usually located on the edge of the image. So, the principle that emphasizes less on the far region and more on the close region was adopted [168]

5.11 Weakly supervised object detection

weakly supervised object detection plays a vital role for developing new systems for computer vision and has received much attention in the past decade. the goal of Weakly supervised object detection is to learn precise object detectors with image category labels [169]. A spatial-temporal network was proposed in [170] to predict saliency in 360° video automatically. This network is a weakly supervised trained which made specially for 360° viewing sphere. an effective Cube Padding (CP) technique was proposed as follows. firstly, a perspective projection was used to render the 360° sphere on six faces of a cube. Then, all six faces were concatenated in convolution, pooling, LSTM layers while using the connectivity between cube's faces.

5.12 Semi-supervised object detection

Semi-supervised learning (SSL) has attracted attention in recent years because it has a potential to improve the performance of models using unlabeled data. In [171], a simple SSL framework called STAC was proposed for object detection. There are two stages of training, the first stage includes training the model using all labeled data, and the second stage includes training the model using both labeled and unlabeled data and generate pseudo labels of unlabeled data. Then, apply data augmentations and augment pseudo labels to unlabeled images. Finally, Compute supervised loss and unsupervised loss to train a detector.

5.13 Omni supervised object detection

Most of the semantic-based frameworks work with pinhole cameras and images with narrow Field-of-View (FoV). However, when working with omnidirectional imagery, Significant decrease in accuracy occurs. In [172], an omnisupervised learning framework was proposed for efficient CNNs. In the preparation stage, annotations were generated for unlabeled panoramas by utilizing a teacher architecture and assembling the prediction of teacher models. In the training phase, the multi-source pinhole images were blended with manually annotated labels and panoramas images with automatically generated labels. The proposed ERF-PSPNET is efficient and suitable for omnidirectional semantic segmentation.

The performance evaluation and elapsed time of different deep learning models for SOD in different SOD datasets is shown in Table 4.

Table 4 Performance evaluation and elapsed time of deep learning models for SOD

#	Model	Year	Dataset	MAE	F-MEASURE	S-MEASURE	Time per frame (s)
1	Fast Yolo [115]	2019	PASCAL VOC	-	0.552	-	0.025
2	UCNet-ABP [57]	2020	DUTS-TE	0.034	-	0.890	0.05
3	PoolNet (VGG-16) [26]	2019	DUTS-TE	0.036	0.892	-	0.03
4	BASNet [56]	2019	DUTS-TE	0.047	-	0.876	0.04
5	HED (VGG-16) [173]	2018	DUTS-TE	0.060	0.800	0.827	0.02
6	SLIC+ VGG-16 [174]	2016	DUTS-TE	0.092	0.747	0.749	0.5
7	HED (ResNet-101) [33]	2018	DUTS-TE	0.065	0.813	0.812	0.5
8	C2S-Net (VGG-16) [175]	2018	DUTS-TE	0.062	0.811	0.822	0.03
9	MSR [158]	2017	DUTS-TE	0.062	0.824	-	0.6
10	PoolNet (VGG-16) [26]	2019	PASCAL-S	0.065	0.88	-	0.03
11	CPD-R (2t50) [25]	2019	PASCAL-S	0.072	0.824	-	0.015
12	DSS (Res2Net-50) [27]	2019	PASCAL-S	0.099	0.841	-	0.149
13	BMPM [176]	2018	PASCAL-S	0.074	0.862	-	0.045
14	HED (VGG-16) [173]	2018	PASCAL-S	0.102	0.830	0.798	0.02
15	C2S-Net (VGG-16) [175]	2018	PASCAL-S	0.081	0.845	0.839	0.03
16	SLIC+ VGG-16 [174]	2016	PASCAL-S	0.122	0.772	0.757	0.5
17	HED (ResNet-101) [33]	2018	PASCAL-S	0.101	0.821	0.796	0.5
18	BMPM [176]	2018	HKU-IS	0.038	0.920	-	0.045
19	UCNet-CVAE [57]	2020	HKU-IS	0.026	-	0.921	0.06
20	HED (VGG-16) [173]	2018	HKU-IS	0.047	0.910	0.884	0.02
21	PoolNet (VGG-16) [26]	2019	HKU-IS	0.03	0.935	-	0.03
22	CPD-R (ResNet50) [25]	2019	HKU-IS	0.034	0.891	-	0.016
23	C2S-Net (VGG-16) [175]	2018	HKU-IS	0.047	0.897	0.886	0.03
24	DSS (Res2Net-50) [27]	2019	HKU-IS	0.05	0.905	-	0.149
25	SLIC+ VGG-16 [174]	2016	HKU-IS	0.072	0.843	0.823	0.5
26	HED (ResNet-101) [33]	2018	HKU-IS	0.050	0.900	0.878	0.5
27	HED (VGG-16) [173]	2018	ECSSD	0.060	0.915	0.886	0.02
28	SLIC+ VGG-16 [174]	2016	ECSSD	0.082	0.865	0.839	0.5
29	HED (ResNet-101) [33]	2018	ECSSD	0.048	0.928	-	0.5
30	MS-FCN (VGG-16) [85]	2016	ECSSD	0.080	0.896	0.080	0.5
31	C2S-Net (VGG-16) [175]	2018	ECSSD	0.057	0.909	0.891	0.03
32	DHSNet (VGG-16) [177]	2016	ECSSD	0.062	0.905	0.884	0.04
33	PiCANet (VGG-16) [178]	2018	ECSSD	0.048	0.932	0.914	0.178

Table 4 (continued)

#	Model	Year	Dataset	MAE	F-MEASURE	S-MEASURE	Time per frame (s)
34	NLDF (VGG-16) [179]	2017	ECSSD	0.063	0.905	0.875	0.08
35	RFCN [180]	2016	ECSSD	0.097	0.898	0852	1.5
36	BMPM [176]	2018	ECSSD	0.044	0.928	-	0.045
37	Amulet (VGG-16) [181]	2017	ECSSD	0.062	0.911	0.894	0.062

6 Experimental results and discussion

In this section, the best used design choices are suggested to help in the future.

6.1 Block-based vs. region-based

In block-based methods there are some shortcomings such as, when the blocks size is large, the boundary of the salient object is not well-preserved. Another shortcoming appears in case of edges with high contrast which usually stands out and this affects the salient object detection. Therefore, the need to use the region-based methods appeared to overcome these shortcomings. Region-based methods have led to a significant evolution in the field of salient object detection, and this is due to several reasons including: the regions number are smaller than number of pixels or blocks and this reduces the computational complexity, and the intrinsic cues such as shapes may be missed in the block-based, but the region-based methods preserve them. As a result, the block-based methods, AC coefficient based [182], achieves 90.19% segmentation accuracy and Histogram based [182], achieves 91.67% segmentation accuracy. While the region-based method, meanshift [183] achieves 96.31% in terms of the best performance index. So, using Region-based methods is the best choice.

6.2 Intrinsic vs. extrinsic

The use of intrinsic cues is popular than extrinsic cues. the intrinsic cues effectiveness has been validated earlier while the extrinsic cues are still less explored. In [184], the saliency detection is applied using method on single image and co-saliency method. On single image, the cluster-based method in [146] achieves F-measure=0.854 while RC [185] achieves F-measure = 0.805. However, when using Co-saliency dataset, the co-saliency method achieves F-measure=0.813. When the co-saliency pairs dataset contains images with clear foreground this will make the second image useless. So, the extrinsic cues may not be effective in some cases, and it is not confirmed to be used in all salient object detection problems.

6.3 Deep learning vs feature-based

In recent years, the deep learning methods have attracted attention in the field of SOD because of the Unprecedented results. In term of Feature-based, HOG, achieves accuracy of 46.0%, 81.48% recall, and 58.80% F-measure. While the deep learning based Fast Yolo achieves accuracy of 76.13%, 43.28% recall, and 55.23% F-measure [115]. in addition, more CNN-based models such as ResNet, VGGNet, Res2Net achieves high performance. In term of MAE, the VGG-16 [178], ResNet-50 [186], Res2Net-50 [27], ResNet-101 [33], achieves 0.048, 0.043, 0.05, 0.065 respectively. While in term of F-measure the results equal 0.932, 0.921, 0.905, 0.813, respectively.

In [57], The proposed CVAE-based and ABP-based models achieve state-of-the-art performance. The CVAE-based model achieves .026 MAE, 0.919 F-MEASURE, and 0.921 S-MEASURE. The ABP-based model achieves 0.027 MAE, 0. 913 F-MEASURE, and 0.917 S-MEASURE.

In [26], the proposed PoolNet model can outperform all former state-of-the-art approaches on six widely used SOD benchmarks. On ECSSD dataset, the model achieves 0.945 in terms

of F-MEASURE and 0.038 in terms of MAE. While on HKU-IS dataset, the model achieves 0. 935 in terms of F-MEASURE and 0.030 in terms of MAE.

In [56], experimental results show that the BASNet model outperforms 15 state-of-the-art methods on six datasets. On ECSSD dataset, the BASNet model achieves 0.942 in terms of F-MEASURE, and 0.037 in terms of MAE.

In [173], The HED architecture takes 0.022 second which is much faster than other methods. It achieves 0.913 in terms of F-MEASURE and 0.045 in terms of MAE on HKU-IS dataset.

In[174], both high level features and low-level features are used for saliency detection which improve the performance. On PASCAL-S dataset, the proposed model achieves 0.771 in terms of F-MEASURE, and 0.121 in terms of MAE. On ECSSD dataset, the proposed model achieves 0.867 in terms of F-MEASURE, and 0.080 in terms of MAE.

In [175], C2S-Net model achieves state-of-the-art result on five popular SOD datasets. on ECSSD, PASCAL-S, HKU-IS, DUTS-TE, and DUT-OMRON, C2S-Net decreases the MAE by 8.5%, 11.9%, 5.9%, 3.1%, and 4.1% respectively. The F-MEASURE is improved by 1.2%, 4.4%, 0.1%, 0.2%, and 2.7%, on ECSSD, PASCAL-S, HKU-IS, DUTS-TE, and DUT-OMRON, respectively.

In [176], the proposed bi-directional message passing method for salient object detection surpasses 13 state-of-the-art methods. On ECSSD dataset, the F-MEASURE is 0.928, and MAE is 0.044. On HKU-IS dataset, the F-MEASURE is 0. 920, and MAE is 0.038.

From the previous evaluation, it observes that the deep learning models have achieved competitive results. The deep learning models will have a promising future in many computer vision problems especially SOD and other models may be discovered in the future.

7 Conclusion

This paper presents the techniques which are used in the salient object detection field. These techniques are divided into two categories: the first one is the Feature-based models, and the other is deep learning-based models. The object detection application has gradually become extensive with the powerful object detectors in many fields such as medical field, plant field, remote sensing field, and transportation field. Various studies which have been recently implemented in this field were reviewed and analyzed. These studies have shown a significant superiority of deep learning techniques in terms of accuracy and performance. It has been demonstrated that CNN, Deep Neural Network, and faster RCNN have been used repetitively for many robust systems of object detection and have obtained contemporary performance in many of these studies on various datasets. Nevertheless, there are several issues that need to be addressed in the future such as panoramic panoptic segmentations which will improve the performance of salient object detection, but there is very little research in this area, and it will be the direction of future research. Also, most of the current research deals with simple background images however images with complex and cluttered backgrounds still need further study. The RGB-D methods are a promising field which still has some challenges in terms of low-quality depth or Incomplete depth maps and the RGB-D datasets for SOD are too small so, new large-scale RGB-D datasets are required for future research.

Funding Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB).

References

1. Goswami A and Dixit M (2020) "An Analysis of Image Segmentation Methods for Brain Tumour Detection on MRI Images," in *9th IEEE International Conference on Communication Systems and Network Technologies*
2. Abdusalomov A, Mukhiddinov M, Djuraev O, Djuraev O and Whangbo TK (2020) "Automatic Salient Object Extraction Based on Locally Adaptive Thresholding to Generate Tactile Graphics," in *Appl. Sci.*
3. Borji A, Cheng MM, Hou Q, Jiang H and Li J (2019) "Salient object detection: A survey," in *Computational Visual Media*
4. Chen K, Wang Y, Hu C and Sh H (2020) "SALIENT OBJECT DETECTION WITH BOUNDARY INFORMATION," in *2020 IEEE International Conference on Multimedia and Expo (ICME)*
5. Zhou LQ, Wang JY, Yu SY, Wu GG, Wei Q, Deng YB, Wu XL and Cui XW (2019) "Artificial intelligence in medical imaging of the liver," in *World J Gastroenterol*, .
6. I. Aganj, M. G. Harisinghani, R. Weissleder and B. Fischl (2018) "Unsupervised Medical Image Segmentation Based on the Local Center of Mass," in *scientific reports*
7. T. Sakinis, F. Milletari, H. Roth, P. Korfiatis, P. Kostandy, K. Philbrick, Z. Akku, Z. Xu, D. Xu and B. J. Erickson (2019) "Interactive segmentation of medical images through fully convolutional neural networks, " in *arXiv e-prints*
8. Voronin V, Semenishchev E, Pismenskova M, Balabaeva O, Agaian S (2019) Medical image segmentation by combing the local, global enhancement, and active contour model. In: *Anomaly Detection and Imaging with X-Rays (ADIX) IV*. Baltimore, Maryland, United States
9. A. Borji, M.-M. Cheng, H. Jiang and J. Li (2015) "Salient Object Detection: A Benchmark," in *IEEE Transactions on Image Processing*
10. Tsai C-C, Yang Y-H, Lin H-W, Wu B-X, Chang EC, Liu HY, Lai J-S, Chen PY, Lin J-J, Chang JS, Wang L-J, Kuo TT, Hwang J-N, Guo J-I (2020) THE 2020 EMBEDDED DEEP LEARNING OBJECT DETECTION MODEL COMPRESSION COMPETITION FOR TRAFFIC IN ASIAN COUNTRIES. In: *2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. United Kingdom, London
11. Arif JMCTM, Niessen WJ, Schoots IG, Veenland JF (2020) Automated Classification of Significant Prostate Cancer on MRI: A Systematic Review on the Performance of Machine Learning Applications, *Cancers (Basel)*, pp 1–13
12. Y. Shen, N. Wu, J. Phang, J. Park, K. Liu, S. Tyagi, L. Heacock, S. G. Kim, L. Moy, K. Cho and K. J. Geras (2020) "An interpretable classifier for high-resolution breast cancer screening images utilizing weakly supervised localization," *arXiv*
13. Papandrianos N, Papageorgiou E, Anagnostis A, Papageorgiou K (2020) Efficient Bone Metastasis Diagnosis in Bone Scintigraphy Using a Fast Convolutional Neural Network Architecture, *Diagnostics (Basel)*, pp 1–17
14. H. Chen, KailaiZhang, P. Lyu, H. Li, LudanZhang, JiWu and C.-H. Lee (2019) "A deep learning approach to automatic teeth detection and numbering based on object detection in dental periapical flms," *Scientific Reports,* pp. 1-19
15. L. Wang, L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin and X. Ruan (2017) "Learning to Detect Salient Objects with Image-level Supervision," in *CVPR2017*
16. L. Hou, Chen-Ping, Y. Hoai and D. Samaras (2016) "Large-scale training of shadow detectors with noisilyannotated shadow examples," in *ECCV*

17. J. Wang, X. Li, L. Hui and J. Yang (2018) "Stacked Conditional Generative Adversarial Networks for Jointly Learning Shadow Detection and Shadow Removal," in *CVPR*
18. J. Zhu, K. G. G. Samuel, S. Z. Masood and M. F. Tappen (2010) "Learning to recognize shadows in monochromatic natural images," in *CVPR*
19. D.-P. Fan, M.-M. Cheng, J.-J. Liu, S.-H. Gao, Q. Hou and A. Borji (2018) "A. Salient objects in clutter: Bringing salient object detection to the foreground," in *the European Conference on Computer Vision (ECCV)*
20. C. Yang, L. Zhang, H. Lu, X. Ruan and M.-H. Yang (2013) "Saliency Detection via Graph-Based Manifold Ranking," in *IEEE Conference on Computer Vision and Pattern Recognition*
21. Li G and Yu Y (2015) "Visual Saliency Based on Multiscale Deep Features," in *IEEE Conference on Computer Vision and Pattern Recognition*
22. J. Shi, Q. Yan, L. Xu and J. Jia (2015) "Hierarchical image saliency detection on extended cssd," in *IEEE Trans. Pattern Anal. Mach. Intell.*
23. Y. Li, X. Hou, C. Koch, J. M. Rehg and A. L. Yuille (2014) "The secrets of salient object segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*
24. Movahedi V and Elder JH (2010) "Design and Perceptual Validation of Performance Measures for Salient Object Segmentation," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*
25. Wu Z, Su L and Huang Q (2019) "Cascaded Partial Decoder for Fast and Accurate Salient Object Detection," *CVPR,* pp. 3907-3916
26. J.-J. Liu, Q. Hou, M.-M. Cheng, J. Feng and J. Jiang (2019) "A Simple Pooling-Based Design for Real-Time Salient Object Detection," *arXiv:1904.09569v1*
27. S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang and P. Torr (2019) "Res2Net: A New Multi-scale Backbone Architecture," *arXiv:1904.01169v2,* pp. 1-10
28. A. Ahmed, A. Jalal and A. A. Rafique (2019) "Salient Segmentation based Object Detection and Recognition using Hybrid Genetic Transform," in *International Conference on Applied and Engineering Mathematics*
29. W. Wang, J. Shen, X. Dong and A. Borji (2018) "Salient Object Detection Driven by Fixation Prediction," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*
30. Alzahrani AJ and Afridi H (2019) "Salient Object Detection: A Distinctive Feature Integration Model," *ArXiv,* Vols. @article{Alzahrani2019SalientOD
31. Mansourian L, Abdullah MT, Abdullah LN, Azman A, Mustaffa MR (2016) A Salient Based Bag of Visual Word Model (SBBoVW) : Improvements toward Difficult Object Recognition and Object Location in Image Retrieval. *KSII Transactions on Internet and Information Systems (TIIS)* 10:769–786
32. Wang Q, Zhang L, Li Y, Kpalma K (2020) Overview of deep-learning based methods for salient object detection in videos. *Pattern Recognition* 104:1–16
33. M.-M. C. X.-W. H. A. B. Z. T. P. T. Qibin Hou (2018) "Deeply Supervised Salient Object Detection with Short Connections," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*
34. P. Pancha, V. C. Raman and S. Mantri (2019) "Plant Diseases Detection and Classification using Machine Learning Models," in *2019 4th International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*
35. P. Prihandoko, B. Bertalya and L. Setyowati (2020) "City Health Prediction Model Using Random Forest Classification Method," in *2020 Fifth International Conference on Informatics and Computing (ICIC)*
36. V. Jackins, S. Vimal, M. Kaliappan and M. Y. Lee (2020) "AI-based smart prediction of clinical disease using random forest classifier and Naive Bayes," *The Journal of Supercomputing*
37. A. Mourad, A. Afifi and A. E. Keshk (2020) "Automated Brain Tumor Segmentation in MRI using Superpixel Over-segmentation and Classification," in *2020 21st International Arab Conference on Information Technology (ACIT)*
38. D. Das, M. Singh, S. S. Mohanty and S. Chakravarty (2020) "Leaf Disease Detection using Support Vector Machine," in *2020 International Conference on Communication and Signal Processing (ICCSP)*
39. Z. Bingzhen, Q. Xiaoming, Y. Hemeng and Z. Zhubo (2020) "A Random Forest Classification Model for Transmission Line Image Processing," in *The 15th International Conference on Computer Science & Education (ICCSE 2020)*
40. A. Aruraj, A. Alex, M. Subathra, N. Sairamya, S. T. George and S. V. Ewards (2019) "Detection and Classification of Diseases of Banana Plant Using Local Binary Pattern and Support Vector Machine," in *2019 2nd International Conference on Signal Processing and Communication (ICSPC)*
41. F. P. Rani, S. Kumar, A. L. Fred, C. Dyson, V. Suresh and P. Jeba (2019) "K-means Clustering and SVM for Plant Leaf Disease Detection and Classification," in *2019 International Conference on Recent Advances in Energy-efficient Computing and Communication (ICRAECC)*

42. Cervantes J, Garcia-Lamont F, Rodríguez-Mazahua L, Lopez A (2020) A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Elsevier*:189–215
43. Yang Z, Li D (2019) "Application of Logistic Regression with Filter in Data Classification," in *2019 Chinese Control Conference (CCC)*. Guangzhou, China
44. D. Tiwari, M. Ashish, N. Gangwar, A. Sharma, S. Patel and S. Bhardwaj (2020) "Potato Leaf Diseases Detection Using Deep Learning," in *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*
45. K. Ahmed, T. R. Shahidi, S. M. I. Alam and S. Momen (2019) "Rice Leaf Disease Detection Using Machine Learning Techniques," in *2019 International Conference on Sustainable Technologies for Industry 4.0 (STI)*
46. J. Singh, S. Bagga and R. Kaur (2020) "Software-based Prediction of Liver Disease with Feature Selection and Classification Techniques," in *International Conference on Computational Intelligence and Data Science (ICCIDS 2019)*
47. A. T. A. Al-khayyat and A. Ibrahim (2020) "Robot Pathplanning By Image Segmentation Using The Fuzzy C-Means Algoritm And KNN Algorithm," in *2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*
48. A. S. Tulshan and N. Raul (2019) "Plant Leaf Disease Detection using Machine Learning," in *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*
49. Ehsani R, Drabløs F (2020) Robust Distance Measures for kNN Classification of Cancer Data. *Cancer Informatics*:1–9
50. J.-e. Liu and F.-P. An, "Image Classification Algorithm Based on Deep Learning-Kernel Function," *Scientific Programming,* pp. 1-14, 2020.
51. S. Y. Chaganti, I. Nanda, K. R. Pandi, T. G. Prudhvith and N. Kumar (2020) "Image Classification using SVM and CNN," in *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*
52. V. Rachapudi and L. Devi (2020) "Improved convolutional neural network based histopathological image classifcation," *Evolutionary Intelligence (2020)*
53. S. Y. Yadhav, T. Senthilkumar, S. Jayanthy and J. J. A. Kovilpillai (2020) "Plant Disease Detection and Classification using CNN Model with Optimized Activation Function," in *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*
54. M. A. Jasim and J. M. Al-tuwaijari (2020) "Plant Leaf Diseases Detection and Classification Using Image Processing and Deep Learning Techniques," in *2020 International Conference on Computer Science and Software Engineering (CSASE)*
55. Q. Yan, L. Xu, J. Shi and J. Jia (2013) "Hierarchical Saliency Detection," in *IEEE Conference on Computer Vision and Pattern Recognition*
56. Qin X, Zhang Z, Huang C, Gao C, Dehghan M, Jagersand M (2019) BASNet: Boundary-Aware Salient Object Detection. *CVPR*:7479–7489
57. Zhang J, Fan D-P, Dai Y, Anwar S, Saleh F, Aliakbarian S, Barnes N (2020) Uncertainty Inspired RGB-D Saliency Detection *arXiv*:1–16
58. P. Ochs, J. Malik and T. Brox (2014) "Segmentation of moving objects by long term video analysis," in *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*
59. D.-P. Fan, W. Wang, M.-M. Cheng and J. Shen (2019) "Shifting More Attention to Video Salient Object Detection," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA
60. P. Yan, G. Li, Y. Xie, Z. Li, C. Wang, T. Chen and L. Lin (2019) "Semi-Supervised Video Salient Object Detection Using Pseudo-Labels," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*
61. F. Perazzi, J. Pont-Tuset, B. McWilliams, L. V. Gool, M. Gross and A. Sorkine-Hornung (2016) "A Benchmark Dataset and Evaluation Methodology for Video Object Segmentation," in *IEEE CVPR*
62. H. Song, W. Wang, S. Zhao, J. Shen and K.-M. Lam (2018) "Pyramid Dilated Deeper ConvLSTM for Video Salient Object Detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*
63. J. Li, C. Xia and X. Chen (2018) "A benchmark dataset and saliency-guided stacked autoencoders for videobased salient object detection," in *IEEE TIP*
64. Qi J, Du J, Siniscalchi M, Ma X, Lee C-H (2020) On Mean Absolute Error for Deep Neural Network Based Vector-to-Vector Regression. *IEEE Signal Processing Letters*:1485–1489
65. Wong T-T (2020) Linear Approximation of F-measure for the Performance Evaluation of Classification Algorithms on Imbalanced Data Sets. *IEEE Transactions on Knowledge and Data Engineering*:1–12
66. D.-P. Fan, M.-M. Cheng, Y. Liu, T. Li and A. Borji (2017) "Structure-measure: A New Way to Evaluate Foreground Maps," in *International Conference on Computer Vision*

67. Ingole PP, Nandedkar (2019) A Review Paper on Salient Object Detection using Edge Preservation and Multi-Scale Contextual Neural Network. *International Journal of Scientific Development and Research (IJSDR)*:261–265
68. Kong Y, Dun Y, Meng J, Wang L, Zhang W, Li X (2020) A Novel Classification Method of Medical Image Segmentation Algorithm. Medical Imaging and Computer-Aided Diagnosis:107–115
69. H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng and S. Li (2013) "Salient Object Detection: A Discriminative Regional Feature," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*
70. W. Xiao, A. Zaforemska, M. Smigaj, Y. Wang and R. Gaulton (2019) "Mean Shift Segmentation Assessment for Individual Forest Tree Delineation from Airborne Lidar Data," *Remote Sens*
71. Li Z, Zheng Y, Cao L, Jiao L, Zhong Y, Zhang C (2020) A Student's t-based density peaks clustering with superpixel segmentation (tDPCSS) method for image color clustering. *Color Research & Application 45*
72. G. Liu and J. Duan (2020) "RGB-D image segmentation using superpixel and multi-feature fusion graph theory," *Signal, Image and Video Processing (SIViP 14)*
73. Y. Huang, Y. Sugano and Y. Sato (2020) "Improving Action Segmentation via Graph-Based Temporal Reasoning," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*
74. P. M. Jensen, A. B. Dahl and V. A. Dahl (2020) "Multi-Object Graph-Based Segmentation With Non-Overlapping Surfaces," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*
75. Z. Mishra, A. Ganegoda, J. Selicha, Z. Wang, S. R. Sadda and Z. Hu (2020) "Automated Retinal Layer Segmentation Using Graph-based Algorithm Incorporating Deep-learning-derived Information," in *Sci Rep 10*
76. Y. Chen, Y. Li and J. Wang (2020) "Remote Aircraft Target Recognition Method Based on Superpixel Segmentation and Image Reconstruction," *Mathematical Problems in Engineering*
77. Singh NK, Singh NJ, Kumar WK (2020) Image classification using SLIC superpixel and FAAGKFCM image segmentation. *IET Image Processing*:487–494
78. T. Tan, Q. Zeng and K. Xuan (2018) "Integrating Multiscale Contrast Regions for Saliency Detection," in *Pacific Rim International Conference on Artificial Intelligence*
79. Srivastava G, Srivastava R (2019) Salient object detection using background subtraction, Gabor filters, objectness and minimum directional backgroundness. *Journal of Visual Communication and Image Representation* 62:330–339
80. Demirovic D (2019) "An Implementation of the Mean Shift Algorithm," *Image Processing On Line,* p. 251–268
81. N. Fatemi, H. Sajedi and M. E. Shiri (2019) "Fully Unsupervised Salient Object Detection," in *2019 4th International Conference on Pattern Recognition and Image Analysis (IPRIA)*
82. X. Xia, S. Wang, X. Zhang, L. Cui and Z. Zhao (2019) "Rgbd Saliency Object Detection Via Regional Feature Clustering," in *2019 12th International Conference on Intelligent Computation Technology and Automation (ICICTA)*
83. N. Fatemi, H. Sajedi and M. E. Shiri (2018) "Salient Object Detection with Segment Features using Mean Shift Algorithm," in *8th International Conference on Computer and Knowledge Engineering (ICCKE 2018)*
84. X. Shen and Y. Wu (2012) "A unified approach to salient object detection via low rank matrix recovery," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*
85. G. Li and Y. Yu (2016) "Deep Contrast Learning for Salient Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*
86. N. Tong, H. Lu, X. Ruan and M.-H. Yang (2015) "Salient Object Detection via Bootstrap Learning," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*
87. T. Liu, Sun, N. Zheng, Tang and Y. Shum (2007) "Learning to Detect A Salient Object," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*
88. J. Li, M. D. Levine, X. An and H. He (2011) "Saliency Detection Based on Frequency and Spatial Domain Analysis," in *BMVA Press*
89. D. Zhang, H. Fu, J. Han, A. Borji and X. Li (2018) "A Review of Co-Saliency Detection Algorithms: Fundamentals, Applications, and Challenges," *ACM Transactions on Intelligent Systems and Technology*
90. H. Li and K. N. Ngan (2011) "A Co-Saliency Model of Image Pairs," in *EEE Transactions on Image Processing*
91. D.-P. Fan, Z. Lin, G.-P. Ji, D. Zhang, H. Fu and M.-M. Cheng, "Taking a Deeper Look at Co-Salient Object Detection," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 2916-2926, 2020.
92. Wu Y, Su Q, Ma W, Liu S, Miao Q (2020) "Learning Robust Feature Descriptor for Image Registration With Genetic Programming," *IEEE. Access*:39389–39389

93. Dong C, Liu J, Xu F, Liu C (2019) Ship Detection from Optical Remote Sensing Images Using Multi-Scale Analysis and Fourier HOG Descriptor. *Remote Sensing*:1–19
94. Y.-T. Chang and T. K. Shih, "A Deep Learning Approach for Dynamic Object Understanding using SIFT, " in *Twelfth International Conference on Ubi-Media Computing (Ubi-Media)*, 2019.
95. A. V. Murthy, B. Rajeshwari and BB, "SOC for image processing using SIFT accelerator," in *IEEE 16th India Council International Conference (INDICON)*, 2019.
96. Wigati EK, Kusuma GP, Utomo Y (2020) Combination of Salient Object Detection and Image Matching for Object Instance Recognition. *Advances in Science, Technology and Engineering Systems Journal* 5: 584–591
97. Z. Abbas, Mu R. S. N. and S. D. Rizvi, "An Efficient Gray-Level Co-Occurrence Matrix (GLCM) based Approach Towards Classification of Skin Lesion," in *2019 Amity International Conference on Artificial Intelligence (AICAI)*, 2019.
98. Y. Adhitya, S. W. Prakosa, and J.-S. Leu, "Feature Extraction for Cocoa Bean Digital Image Classification Prediction for Smart Farming Application," *Agronomy,* 2020.
99. A. Ramola, A. K. Shakya and D. V. Pham, "Study of statistical methods for texture analysis and theirmodern evolutions," *Engineering Reports,* 2020.
100. Jain I, Shah SS, Gopalakrishnan T (2020) "Implementation of Local Binary Pattern in Feature Recognition and Novel Approach to Classify the Images Based on TextureFeatures," *International Journal of Advanced*. *Science and Technology*:8013–8025
101. Bi Y, Xue B, Zhang M (2020) An Effective Feature Learning Approach Using Genetic Programming With Image Descriptors for Image Classification. *IEEE Computational Intelligence Magazine*:65–77
102. S. Ghaffari, P. Soleimani, K. F. Li and D. Capson, "FPGA-based Implementation of HOG Algorithm: Techniques and Challenges," in *2019 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM)*, 2019.
103. P. Gupta, P. Gupta and P. Gupta, "Evaluation of Different Feature Descriptor Algorithms on Classification Task," in *2019 5th International Conference On Computing, Communication, Control And Automation (ICCUBEA)*, 2019.
104. R. Nuari, E. Utami and S. Raharjo, "Comparison of Scale Invariant Feature Transform and Speed Up Robust Feature for Image Forgery Detection Copy Move," in *2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, 2019.
105. A. S. Jubair, A. J. Mahna and A. J. Mahna, "Scale Invariant Feature Transform Based Method for Objects Matching," in *2019 International Russian Automation Conference (RusAutoCon)*, 2019.
106. Gupta S, Thakur K, Kumar M (2020) 2D-human face recognition using SIFT and SURF descriptors of face's feature regions. *Visual Computer*
107. F. Hidalgo and T. Bräunl (2020) "Evaluation of Several Feature Detectors/Extractors on Underwater Images towards vSLAM," *Sensors (Basel),* pp. 1-16
108. S. Ke-Chen, Y. Yun-Hui, C. Wen-Hui and Z. Xu (2013) "Research and Perspective on Local Binary Pattern," in *Acta Automatica Sinica*
109. M. Calonder, V. Lepetit, C. Strecha and P. Fua, "BRIEF: Binary Robust Independent Elementary Features, " in *Computer Vision - ECCV 2010, 11th European Conference on Computer Vision*, 2010.
110. E. Rublee, V. Rabaud, K. KonoligeGary and B. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *IEEE International Conference on Computer Vision*, 2011.
111. V. A. A. S. Rao, V. S. Shekhar, A. Kumar, C.K.N. Balasubramany and Murthya S. Natarajan, "Feature Extractionusing ORB-RANSAC for Face Recognition," in *Proceedings of the 4th International Conference on Eco-friendly Computing and Communication Systems*, 2015.
112. BoyrazErtuğrul P, Bayraktar B (2017) Analysis of feature detector and descriptor combinations with a localization experiment for various performance metrics. *Turkish Journal of Electrical Engineering and Computer Sciences*:2444–2454
113. Q. Tian, B. Zhou, W.H. Zhao, Y. Wei and W.-w. Fei, "Human Detection using HOG Features of Head and Shoulder Based on Depth Map," *ACADEMY PUBLISHER,* pp. 2223-2230, 2013.
114. Mohammed MG, Melhum AI (2020) Implementation of HOG Feature Extraction with Tuned Parameters for Human Face Detection. *International Journal of Machine Learning and Computing*:654–661
115. F. Benjelloun, K. Abbad, M. A. Sabri, A. Aarab and A. Yahyaouy, "Comparison of two vehicle detection methods based on the oriented gradients histogram and the deep neural network," in *2019 International Conference on Intelligent Systems and Advanced Computing Sciences (ISACS)*, 2019.
116. J. Shang, C. Chen, H. Liang and H. Tang (2016) "Object recognition using rotation invariantlocal binary pattern of significant bit planes," *IET Image Processing*
117. M. Waqas and N. Fukushima (2020) "Comparison of Image Features Descriptions for Diagnosis of Leaf Diseases," in *APSIPA Annual Summit and Conference 2020*

118. Farooq J (2016) "Object Detection and Identification using SURF and BoW Model," in *Conference: International conference on Computing, Electronic and Electrical Engineering (ICE Cube)*
119. H. Kuang, K.-F. Yang, L. Chen, Y.-J. Li, L. L. H. Chan and H. Yan, "Bayes Saliency-Based Object Proposal Generator for Nighttime Traffic Images," in *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*, 2018.
120. E. Romera, L. M. Bergasa, K. Yang, J. M. Alvarez and R. Barea, "Bridging the Day and Night Domain Gap for Semantic Segmentation," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, 2019.
121. G. P.G., "Different Approaches for Semantic Segmentation," in *2020 5th International Conference on Communication and Electronics Systems (ICCES)*, COIMBATORE, India, 2020.
122. Ferariu L, Mihai M (2020) "CNN-Based Cascade with Skipping Connections for Semantic Segmentation," in *2020 International Symposium ELMAR*. Zadar, Croatia
123. Y. Liu, M.-M. Cheng, X. Hu, J.-W. Bian, L. Zhang, X. Bai and A. J. Tang, "TRADITIONAL METHOD INSPIRED DEEP NEURAL NETWORK FOR EDGE DETECTION," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI),* pp. 1-8, 2019.
124. B. Jiang, X. Li, L. Yin, W. Yue and S. Wang, "Object Recognition in Remote Sensing Images Using Combined Deep Features," in *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, Chengdu, China, 2019.
125. Zhang Q, Shi Y, Zhang X (2020) Attention and boundary guided salient object detection. *Pattern Recognition* 107:1–12
126. Khan A, Sohail A, Zahoora U (2020) A survey of the recent architectures of deep convolutional neural networks. *Artifcial Intelligence Review 53*:5455–5516
127. Üreten K, Erbay H, Maraş HH (2020) Detection of rheumatoid arthritis from hand radiographs using a convolutional neural network. *Clin Rheumatol*:969–974
128. Y. Jiang and C. Li, "Convolutional Neural Networks for Image-Based HighThroughput Plant Phenotyping: A Review," *Plant Phenomics,* pp. 1-22, 2020.
129. Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," in *Proceedings of the IEEE*, 1998.
130. A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *25th International Conference on Neural Information Processing Systems*, 2012.
131. K. Simonyan and A. Zisserman, "VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION," in *arXiv*, 2015.
132. K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," in *arXiv*, 2015.
133. Y. Qian, H. Zheng, D. He, Z. Zhang and Z. Zhang, "R-CNN Object Detection Inference With Deep Learning Accelerator," in *2018 IEEE/CIC International Conference on Communications in China (ICCC Workshops)*, 2018.
134. L. Sommer, N. Schmidt, A. Schumann and J. Beyerer, "Search Area Reduction Fast-RCNN for Fast Vehicle Detection in Large Aerial Imagery," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, 2018.
135. Xiao Y, Wang X, Zhang P, Meng F, Shao F (2020) Object Detection Based on Faster R-CNN Algorithm with Skip Pooling and Fusion of Contextual Information. *Sensors*:1–20
136. S. Shivajirao, R. Hantach, S. B. Abbes and P. Calvez, "Mask R-CNN End-to-End Text Detection and Recognition," in *2019 18th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2019.
137. Y. Xiao, Z. Tian, J. Yu, Y. Zhang, S. Liu and Shaoyi, "A review of object detection based on deep learning," in *Multimedia Tools and Applications*, 2020.
138. F. Ucar, B. Dandil and F. Ata (2020) "Aircraft Detection System Based on Regions with Convolutional Neural Networks," *INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING (ijisae),* pp. 147-153.
139. K. Wei, B. Wang and J. Saniie, "Faster Region Convolutional Neural Networks Applied to Ultrasonic Images for Breast Lesion Detection and Classification," in *2020 IEEE International Conference on Electro Information Technology (EIT)*, Chicago, IL, USA, 2020.
140. Alalharith DM, Alharthi HM, Alghamdi WM, Alsenbel YM, Aslam N, Khan IU, Shahin SY, Dianišková S, Alhareky MS, Barouch KK (2020) "A Deep Learning-Based Approach for the Detection of Early Signs of Gingivitis in Orthodontic Patients Using Faster Region-Based Convolutional Neural Networks," *Environmental Research and Public. Health*:1–10
141. Li D, Wang R, Xie C, Liu L, Zhang J, Li R, Wang F, Zhou M, Liu W (2020) A Recognition Method for Rice Plant Diseases and Pests Video Detection Based on Deep Convolutional Neural Network. *Sensors*:1–21
142. Y. Zhang, C. Song and D. Zhang, "Deep Learning-Based Object Detection Improvement for Tomato Disease," *IEEE Access,* 2020.

143. A. Tourani, S. Soroori, A. Shahbahrami, S. Khazaee and A. Akoushideh, "A Robust Vehicle Detection Approach based on Faster R-CNN Algorithm," in *2019 4th International Conference on Pattern Recognition and Image Analysis (IPRIA)*, Tehran, Iran, 2019.
144. Z. Dai, E. Carver, C. Liu, "Segmentation of the Prostatic Gland and the Intraprostatic Lesions on Multiparametic Magnetic Resonance Imaging Using Mask Region-Based Convolutional Neural Networks," *Advances in Radiation Oncology,* pp. 473-481, 2020.
145. Liu L, Ouyang W, Wang X, Fieguth P, Chen J, Liu X, Pietikäinen M (2020) Deep Learning for Generic Object Detection: A Survey. *International Journal of Computer Vision*:261–318
146. M. R, A. N, D. R and R. R, "Enhanced Missing Object Detection System using YOLO," in *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Coimbatore, India, 2020.
147. G. Sha, J. Wu and B. Yu, "Detection of Spinal Fracture Lesions based on Improved Yolov2," in *2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, Dalian, China, 2020.
148. C. Kumar, P. R Mohana, "YOLOv3 and YOLOv4: Multiple Object Detection for Surveillance Applications," in *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Tirunelveli, India, 2020.
149. Y. Li and C. Lv, "SS-YOLO: An Object Detection Algorithm based on YOLOv3 and ShuffleNet," in *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, Chongqing, China, 2020.
150. J.-X. Zhao, J.-J. Liu, D.-P. Fan, Y. Cao, J.-F. Yang and M.-M. Cheng, "EGNet: Edge Guidance Network for Salient Object Detection," in *IEEE International Conference on Computer Vision (ICCV) 2019*, 2019.
151. Mandal V, Mussah AR, Jin P, Adu-Gyamf Y (2020) Artificial Intelligence-Enabled Traffic Monitoring System. *Sustainability*:1–21
152. Cao C, Wu J, Zeng X, Feng Z, Wang T, Yan X, Wu Z, Wu Q, Huang Z (2020) Research on Airplane and Ship Detection of Aerial Remote Sensing Images Based on Convolutional Neural Network. *MDPI*:1–16
153. K. Zhao and X. Ren, "Small Aircraft Detection in Remote Sensing Images Based on YOLOv3," in *IOP Conference Series: Materials Science and Engineering*, 2019.
154. H. Peng, B. Li, W. Xiong, W. Hu and R. Ji, "RGBD Salient Object Detection: A Benchmark and Algorithms," in *ECCV 2014*, 2014.
155. X. Hu, K. Yang, L. Fei and K. Wang, "ACNET: ATTENTION BASED NETWORK TO EXPLOIT COMPLEMENTARY FEATURES FOR RGBD SEMANTIC SEGMENTATION," in *IEEE International Conference on Image Processing (ICIP 2019)*, 2019.
156. W.-D. Jin, J. Xu, Q. Han, Y. Zhang and M.-M. Cheng, "CDNet: Complementary Depth Network for RGB-D Salient Object Detection," in *IEEE TRANSACTIONS ON IMAGE PROCESSING*, 2021.
157. B. Hariharan, P. Arbel´aez, R. Girshick and J. Malik, "Simultaneous Detection and Segmentation," in *ECCV*, 2014.
158. G. Li, Y. Xie, L. Lin and Y. Yu, "Instance-Level Salient Object Segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
159. Q. Wang, Y. Yuan, P. Yan and X. Li, "Saliency Detection by Multiple-Instance Learning," in *IEEE Transactions on Cybernetics*, 2013.
160. A. Kirillov, K. He, R. Girshick, C. Rother and P. Dollar, "Panoptic Segmentation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
161. A. Jaus, K. Yang and R. Stiefelhagen, "Panoramic Panoptic Segmentation: Towards Complete Surrounding Understanding via Unsupervised Contrastive Learning," *ArXiv,* 2021.
162. A. Vaswani, N. Shazeer, N. Parmar and J. Uszkoreit, "Attention is all you need," in *31st Conference on Neural Information Processing Systems*, 2017.
163. N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov and S. Zagoruyko, "End-to-End Object Detection with Transformers," in *ECCV2020*, 2020.
164. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo B (2021) Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. *arXiv*
165. J. Wu, W. Zhou, T. Luo, L. Yu and J. Lei, "Multiscale multilevel context and multimodal fusion for RGB-D salient object detection," *Signal Processing,* 2021.
166. K. Yang, J. Zhang, S. Reiß, X. Hu and R. Stiefelhagen, "Capturing Omni-Range Context for Omnidirectional Segmentation," in *CVPR*, 2021.
167. S. Goferman, L. Zelnik-Manor and A. Tal, "Context-Aware Saliency Detection," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012.
168. Q. Wang, Y. Yuan and P. Yan, "Visual Saliency by Selective Contrast," in *IEEE Transactions on Circuits and Systems for Video Technology*, 2013.

169. X. Li, M. Kan, S. Shan and X. Chen, "Weakly Supervised Object Detection with Segmentation Collaboration," in *Computer Vision and Pattern Recognition*, 2019.
170. H.-T. Cheng, C.-H. Chao, J.-D. Dong, H.-K. Wen, T.-L. Liu and M. Sun, "Cube Padding for Weakly-Supervised Saliency Prediction in 360° Videos," in *Computer Vision and Pattern Recognition*, 2018.
171. K. Sohn, Z. Zhang, C.-L. Li, H. Zhang, C.-Y. Lee and T. Pfister, "A Simple Semi-Supervised Learning Framework for Object Detection," in *Computer Vision and Pattern Recognition*, 2020.
172. K. Yang, X. Hu, Y. Fang, K. Wang and R. Stiefelhagen, "Omnisupervised Omnidirectional Semantic Segmentation," in *IEEE Transactions on Intelligent Transportation Systems*, 2020.
173. S. Chen, X. Tan, B. Wang and X. Hu, "Reverse Attention for Salient Object Detection," in *ECCV*, 2018.
174. G. Lee, Y.-W. Tai and J. Kim, "Deep Saliency with Encoded Low level Distance Map and High Level Features," in *CVPR*, 2016.
175. X. Li, F. Yang, H. Cheng, W. Liu and D. Shen, "Contour Knowledge Transfer for Salient Object detection, " in *ECCV*, 2018.
176. L. Zhang, J. Dai, H. Lu, Y. He and G. Wang, "A Bi-Directional Message Passing Model for Salient Object Detection," in *CVPR 2018*, 2018.
177. N. Liu and J. Han, "DHSNet: Deep Hierarchical Saliency Network for Salient Object Detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.
178. N. Liu, J. Han and M.-H. Yang, "PiCANet: Learning Pixel-wise Contextual Attention for Saliency Detection," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.
179. Z. Luo, A. Mishra, A. Achkar, J. Eichel, S. Li and P.-M. Jodoin, "Non-Local Deep Features for Salient Object Detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
180. L. Wang, L. Wang, H. Lu, P. Zhang and X. Ruan, "Saliency Detection with Recurrent Fully Convolutional Networks," in *Computer Vision – ECCV 2016*, 2016.
181. P. Zhang, D. Wang, H. Lu, H. Wang and X. Ruan, "Amulet: Aggregating Multi-level Convolutional Features for Salient Object Detection," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.
182. D.Sasirekha and E. Chandra, "Enhanced Techniques for PDF Image Segmentation and Text Extraction," *(IJCSIS) International Journal of Computer Science and Information Security,* 2012.
183. Li J, Chen H, Li G, He B, Zhang Y, Tao X (2015) Salient object detection based on meanshift filtering and fusion of colour information. *IET Image Processing*:1–9
184. H. Fu, X. Cao and Z. Tu, "Cluster-Based Co-Saliency Detection," in *IEEE TRANSACTIONS ON IMAGE PROCESSING*, 2013.
185. M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang and S.-M. Hu, "Global Contrast based Salient Region Detection," in *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, 2011.
186. T. Wang, L. Zhang, S. Wang, H. Lu, G. Yang, X. Ruan and A. Borji, "Detect Globally, Refine Locally: A Novel Approach to Saliency Detection," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.
187. D. A. Klein and S. Frintrop, "Center-surround Divergence of Feature Statistics for Salient Object Detection," in *2011 IEEE International Conference on Computer Vision*, 2011.
188. R. Girshick, "Fast R-CNN," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015.
189. S. Ren, K. He, R. B. Girshick and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," in *NIPS'15: Proceedings of the 28th International Conference on Neural Information Processing Systems*, 2015.
190. K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.
191. J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
192. J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *arXiv*, 2016.
193. J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," in *arXiv*, 2018.