ElDet: An Anchor-free General Ellipse Object Detector

Tianhao Wang[1], Changsheng Lu[2], Ming Shao[3], Xiaohui Yuan[1], and Siyu Xia[1]

[1] School of Automation, Southeast University, Nanjing, China
[2] College of Engineering and Computer Science, The Australian National University, Canberra, Australia
[3] Computer and Information Science Department, University of Massachusetts Dartmouth, Dartmouth, USA
wangtianhao@seu.edu.cn, Changsheng.Lu@anu.edu.au, mshao@umassd.edu, yuanxh@seu.edu.cn, xsy@seu.edu.cn

Abstract. Ellipse detection is a fundamental task in object shape analysis. Under complex environments, the traditional image processing based approaches may under-perform due to the hand-crated features. Instead, CNN-based approaches are more robust and powerful. In this paper, we introduce an efficient anchor-free data-augmentation based general ellipse detector, termed *ElDet*. Different from existing CNN-based methods, our *ElDet* relies more on edge information which could excavate more shape information into learning. Specifically, we first develop an edge fusion module to composite an overall edge map which has more complete boundary and better continuity. The edge map is treated as augmentation input for our *ElDet* for ellipse regression. Secondly, three loss functions are tailored to our *ElDet*, which are angle loss, IoU loss, and binary mask prediction loss to jointly improve the ellipse detection performance. Moreover, we contribute a diverse ellipse dataset by collecting multiple classes of elliptical objects in real scenes. Extensive experiments show that the proposed ellipse detector is very competitive to state-of-the-art methods.

1 Introduction

Geometric shape is a crucial characteristic to objects and is of great importance for object detection and recognition. In some scenarios, we pay more attention to shape & contour information, such as medical imaging diagnosis [1], object counting [2], and CAD workpieces recognition [3,4]. Existing shape detection methods are mainly based on traditional image processing [5,6], which heavily depend on low-level edge lines and omit physical meanings of objects. Thus, such methods are susceptible to interference from extraneous noisy edges or lines.

Ellipse shape commonly appears in various scenes and can be modelled by 5-D parameters. Compared to other shapes, *e.g.*, rectangle, ellipse presents superior information. Traditional ellipse detection algorithms [5,7] usually build the mathematical model based on segmented edges, contours, and curvatures.

These methods have achieved relatively accurate and efficient ellipse detection. However, they require appropriate pre-processing to ensure the quality of ellipse fitting. The most significant pre-processing step used by these methods is edge detection. Current edge detectors will inevitably introduce extraneous lines and noise, which largely affects the subsequent detection. In addition, most of these methods require artificially set some thresholds and other parameters, which is difficult to guarantee the consistency of accuracy and efficiency.

With the rapid development of deep learning, it is apparent that CNN-based methods are more robust and efficient, especially under complex scenes. Recently, approach [8] develops an anchor-based ellipse detector based on Faster R-CNN [9], but it is two-stage and subject to specific object classes. Some researchers suggest that Convolutional Neural Network (CNN) is strongly biased towards textures rather than shapes [10]. Therefore, for class-specific object detection, texture information can indeed play a big role, but for general ellipse detection, edge information is more desirable as it describes the essential geometric shapes.

In addition, multi-task learning (MTL) is important in machine learning [11,12]. Compared to single-task learning (STL), MTL jointly optimize multiple functions with close relevance. During training, the shared and more general representations are learned, thus achieving better generalization during testing. In most of cases, MTL has proved to outperform STL. In this paper, we use the ellipse mask segmentation as a pretext task in addition to ellipse detection task, forming the multi-task learning fashion. The reason behind this is that the edge has close relationship to binary ellipse masks, which both provide shape information crucial to ellipse detection.

We propose a novel data-augmentation based ellipse detector (*ElDet*) in this paper, which is anchor-free, one-stage, and general. Unlike existing CNN-based approaches, we enrich the input with data augmentation by combining edge maps using our *Edge Fusion Module*, which is omitted by EllipseNet [13] and Ellipse R-CNN [14]. The edge information allows our detector to effectively use contour shape information for precise ellipse object detection. Moreover, our model has a potential of detecting various ellipse-shaped objects, regardless of their actual class labels. During regressing the ellipse parameters, elliptic angle is difficult to learn and has a larger impact on rotated object detection. Recent works have proposed new bounding box representations, such as ordered quadrilateral [15]. However, most popular angle regression methods suffer from the problem of boundary discontinuity (*i.e.*, $-\frac{\pi}{2}$ *vs.* $\frac{\pi}{2}$), which may lead a large and unstable loss. To address this issue, we propose an aspect-ratio based angle loss and a Gaussian distribution based IoU loss. Further, we also add an auxiliary task of binary mask segmentation to fully exploit the edge information from ellipse objects. The contributions are summarized as follows:

1. We propose an anchor-free and general ellipse object detector, which is simple, effective, and capable of being applied to downstream tasks such as irregular-shape object detection.
2. We design an edge fusion module that learns adaptive weight coefficients and fuse multiple edge maps. The obtained fused edge maps will be used for

data augmentation. This module and subsequent detectors together consist of an end-to-end model.
3. We design three dedicated losses for precise ellipse detection including aspect-ratio based angle loss, Gaussian distribution based IoU loss, and binary mask regression loss, which jointly optimizes the *ElDet*.
4. A new ellipse detection dataset collected from real-world scenarios is contributed to enrich the diversity of datasets in the field of ellipse detection.

2 Related Works

2.1 Anchor-free Object Detector.

Anchor boxes are widely used in mainstream detectors, such as Faster R-CNN [9] and YOLOv3 [16]. Anchor-based object detectors are mainly based on anchor boxes which can be considered as pre-designed sliding-windows and proposals. With prior knowledge, anchor boxes significantly improves the accuracy and speed of the predicted object bounding boxes. However, the design of anchor boxes is rather tricked and introduces a large number of hyper-parameters.

To avoid the overwhelming exploration of anchor boxes, anchor-free object detectors are proposed. The anchor-free methods are roughly divided into two branches: i) dense prediction approaches, *e.g.*, DenseBox [17] and FCOS [18]; ii) keypoint-based approaches, *e.g.*, CornerNet [19] and CenterNet, which closely relate to keypoint detection approaches [20,21]. Our method falls into anchor-free keypoint-based category.

CenterNet is one-staged object detector. It treats each object as a point, *i.e.*, center point of the object bounding box. The center point can be regarded as a single shape-agnostic anchor. The detector locates the object center point by keypoint estimation and regress other object attributes directly, such as size, direction, 3-D position and even posture. Moreover, CenterNet does not need Non-Maximum Suppression (NMS) [22], since it only has one positive "anchor" per object. With its minimalist design and great performance, CenterNet is used as basic framework for many anchor-free methods.

2.2 Rotated Object Detection.

Classical object detection typically uses a horizontal rectangular box to frame the object position. However, for some scenarios, such as remote sensing object detection and text detection, the object has a rotation angle. Using rotated object detection to precisely locate the object is beneficial to provide more accurate inputs for subsequent tasks such as recognition and analysis. Most rotated object detectors are inherited from traditional anchor-based object detectors by using rotated bounding boxes or quadrangles.

The main problem of rotated object detection is the angle prediction. Gliding Vertex [23] and RSDet [15] represent the bounding box by four corner points and determine the rotation angle by calculating the offset of corner points. CSL [24]

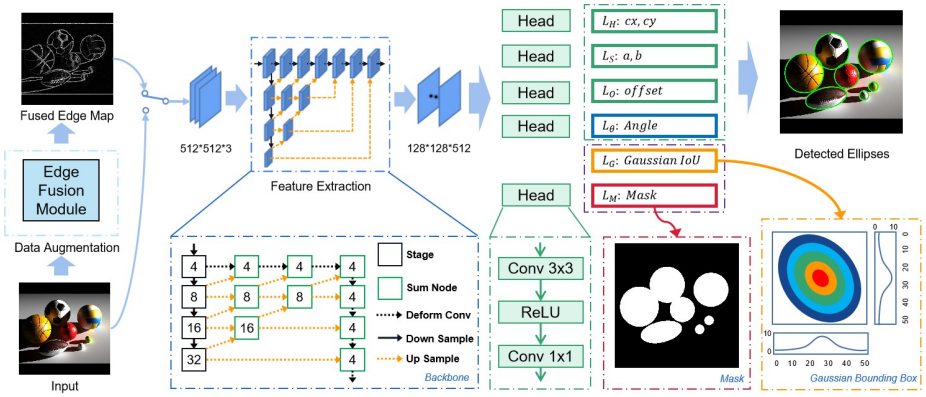


Fig. 1. Overview of the proposed ellipse detector. We use DLA-34 [28] as the backbone for feature extraction. There are two types of losses, where one is for regressing ellipse parameters, and another, namely, mask segmentation, is auxiliary to promote the performance of ellipse detection.

and DCL [25] transform angle prediction from a regression problem to a classification problem. GWD [26] and KLD [27] convert the rotation regression loss into the distance of two 2-D Gaussian distributions. We also follow rotated object detection and introduce the 2-D Gaussian distribution into ellipse detection.

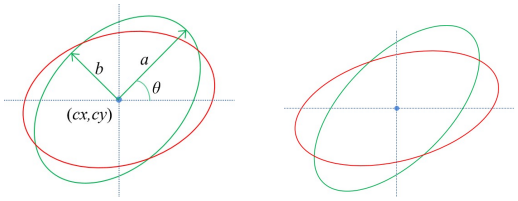
2.3 Ellipse Detection.

Traditional ellipse detection can be roughly divided into three streams, namely, clustering-based, optimization-based, and arc-based methods. Lu *et al.* [7] propose an efficient ellipse detection based on arc support line segment. Recently, CNN-based methods go viral. For instance, Li *et al.* [8] replace the Region Proposal Network (RPN) in Faster R-CNN by a Gaussian Proposal Network (GPN). Ellipse R-CNN [14] presents a two-stage detector based on Mask R-CNN and solves the problem in occluded and cluttered scenes. Apart from these anchor-based detectors, EllipseNet [13] develops an anchor-free ellipse detector in medical images. In this paper, we push further by using edge information and improving the accuracy of angle regression.

3 Proposed Method

3.1 Overview of Framework

As shown in Fig. 1, our framework is developed based on CenterNet [29]. We adapt DLA-34 [28] as the backbone. The goal of the detector is to regress the five parameters of ellipse (x, y, a, b, θ), where (x, y) are the coordinates of the center point, a, b $(a \geq b)$ are the major and minor axis, and $\theta \in [-90, 90)$ is the rotation angle. We introduce edge images as augmented data. To improve



(a) R_{ab}=1.5, IoU=0.78 (b) R_{ab}=2, IoU=0.63

Fig. 2. Influence of aspect ratio R_{ab} on IoU. When the center points overlap, the greater the ratio, the more significant the impact of the angle (The angle of red ellipse is 15°, while the green one is 45°).

the accuracy of angle regression, we add an angle loss based on the aspect ratio, and use Gaussian Wasserstein distance (GWD) loss. In addition, binary mask segmentation is adopted to enhance the network training.

3.2 Ellipse Parameters Regression

We follow CenterNet in the regression of the center point coordinates, the major and minor axis, and the offset. In particular, focal loss function [30] and smooth-L1 loss function are respectively used to optimize (x, y) and a, b. Size Loss $\mathcal{L}_S$ is used to predict a, b. Given an input image $I \in R^{W\times H\times 3}$ with height H and width W, the predicted heatmap $\hat{Y} \in [0,1]^{\frac{W}{R}\times\frac{H}{R}\times 2}$ represents the probability map of the center point location and Y represents the ground truth heatmap. We set output stride $R = 4$. To reduce the discretization error caused by the output stride, the offset is also predicted and offset loss $\mathcal{L}_O$ is calculated with smooth-L1 loss. The heatmap loss $\mathcal{L}_H$ is obtained as follows:

$$\mathcal{L}_H = \frac{-1}{N}\sum_{xyc}\begin{cases}(1-\hat{Y}_{xyc})^{\alpha} log(\hat{Y}_{xyc}), & if \quad Y_{xyc}=1\\ (1-Y_{xyc})^{\beta}(\hat{Y}_{xyc})^{\alpha}\log(1-\hat{Y}_{xyc}), & others\end{cases} \tag{1}$$

where x, y, c indexes a channel, α and β are hyper-parameters of the focal loss, N is the number of keypoints in image I. Moreover, $\alpha = 2$ and $\beta = 4$.

Among the five parameters of ellipse, rotation angle θ largely affects detection results, especially on IoU, as shown in Fig. 2. Assuming (x, y) are equal, the influence of the angle on IoU relies on the aspect ratio $R_{ab} = a/b$. The greater the ratio, the more significant the impact of the angle. This means it is difficult to predict the rotation angle for a round-like object, and too much attention on the angle of round-like object also raises challenges in the angle regression.

To solve this issue, we add a weight based on aspect ratio to angle loss. The angle loss $\mathcal{L}_\theta$ is reformulated as follows:

$$W_\theta = \begin{cases}1 & \text{if } R_{ab} < 1.2\\ 2 & \text{if } R_{ab} \geq 1.2,\end{cases} \tag{2}$$

$$\mathcal{L}_\theta = W_\theta * \text{smooth}_{L1}(\theta_p, \theta_g), \tag{3}$$

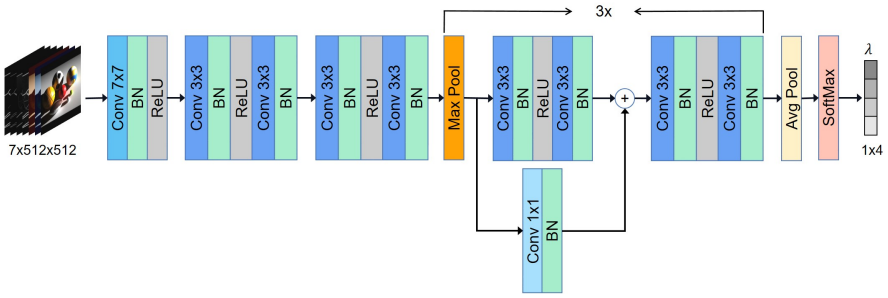


Fig. 3. Network structure of Edge Fusion Module.

where W_θ represents the angle loss weight, θ_p and θ_g are predicted rotation angle value and ground-truth rotation angle value. We use smooth-L1 loss function to calculate the difference between θ_p and θ_g.

3.3 Edge Fusion Module

CNNs are biased towards texture information [10], while ellipse detection relies more on shape information such as contours and edges. Therefore, we introduce edge maps as data augmentation for inputs, and our extensive experiments show that edge maps yield significant improvements for ellipse detection.

Different edge detection methods have their own advantages and yet shortcomings such as sensitivity to noise, positioning, and continuity. In order to obtain high-quality edge maps, we design an adaptive Edge Fusion Module (EFM) for edge maps which are extracted by multiple methods. The EFM is based on ResNet-18[31] with some modifications, as shown in Fig 3.

To obtain an overall better edge map, firstly, we extract four edge maps using *Canny* [32], *Sobel* [33], *Laplacian* [34] operators and *AdaptiveThreshold* [35] method. Then we concatenate the RGB image and the four different edge maps by channel and feed them into the EFM to obtain the weight coefficients corresponding to the four edge maps. The final edge map is obtained via the weighted summation:

$$\begin{aligned} Map &= \sum_{1}^{i=4} \lambda_i \cdot Map_i \\ \sum_{1}^{i=4} \lambda_i &= 1 \end{aligned} \tag{4}$$

where Map means final edge map, Map_i means edge maps extracted by four edge detectors and λ_i are their corresponding weights. The fused edge map will be normalized into $[0, 255]$. With this fusion method, we are able to improve the value of the true edge, reduce the interference of noise, and make the contour more complete (Fig. 4).

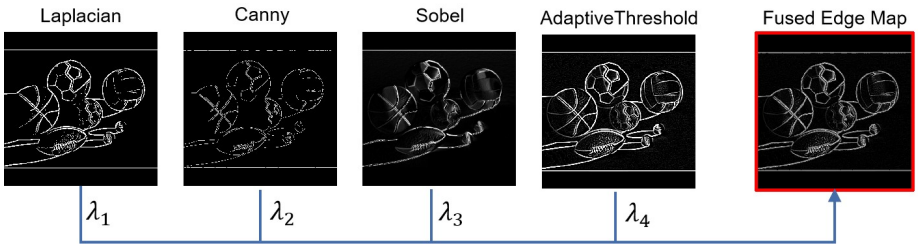


Fig. 4. Weighted summation of edge maps based on the learned weight coefficients produced by Edge Fusion Module (EFM). The left four different edge maps are extracted by *Laplacian*, *Canny*, *Sobel*, and *AdaptiveThreshold*, respectively, and the rightmost is the fused edge map.

Note that we use the edge map as a way of data augmentation, which means that during the training process we input the edge fusion map with probability of 0.5, otherwise we input the RGB image to the subsequent detector.

3.4 Gaussian IoU

As the IoU calculation method for rectangle bounding box is sub-optimal for ellipse detection, we discover that 2-D Gaussian distribution matches ellipse better in shape. In addition, 2-D Gaussian IoU can well integrate into loss functions to address the boundary discontinuity and round-like problem. We simply parameterize a 2-D Gaussian distribution by the mean μ and the covariance matrix Σ in ellipse bounding box $\mathcal{B}(x, y, a, b, \theta)$:

$$\begin{aligned}\Sigma^{\frac{1}{2}} &= RSR^T \\ &= \begin{pmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{pmatrix} \begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix} \begin{pmatrix} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{pmatrix} \\ &= \begin{pmatrix} a \cdot cos^2\theta + b \cdot sin^2\theta & (a-b)cos\theta sin\theta \\ (a-b)cos\theta sin\theta & a \cdot sin^2\theta + b \cdot cos^2\theta \end{pmatrix} \\ \mu &= [x, y] \end{aligned} \tag{5}$$

Further, the Wasserstein distance [36] $\mathbf{W}$ between two Gaussian probabilities $X \sim \mathcal{N}(\mu_1, \mathbf{\Sigma}_1)$ and $Y \sim \mathcal{N}(\mu_2, \mathbf{\Sigma}_2)$ is expressed as:

$$\mathbf{W}(\mu; \nu) := \inf \mathbb{E}(||X - Y||_2^2)^{1/2}. \tag{6}$$

Consequently, we could compute the IoU loss L_G via Wasserstein distance as:

$$\mathcal{L}_G = ||\mu_1 - \mu_2||_2^2 + \mathbf{Tr}(\mathbf{\Sigma}_1 + \mathbf{\Sigma}_2 - 2(\mathbf{\Sigma}_1^{1/2}\mathbf{\Sigma}_2\mathbf{\Sigma}_1^{1/2})^{1/2}), \tag{7}$$

where

$$\mathbf{Tr}((\mathbf{\Sigma}_1^{1/2}\mathbf{\Sigma}_2\mathbf{\Sigma}_1^{1/2})^{1/2}) = \mathbf{Tr}((\mathbf{\Sigma}_2^{1/2}\mathbf{\Sigma}_1\mathbf{\Sigma}_2^{1/2})^{1/2}). \tag{8}$$

Overall, we use IoU loss L_G to measure the error between predicted ellipse and groundtruth ellipse. By optimizing L_G, our model is able to regress ellipse parameters $\mathcal{B}(x, y, a, b, \theta)$.

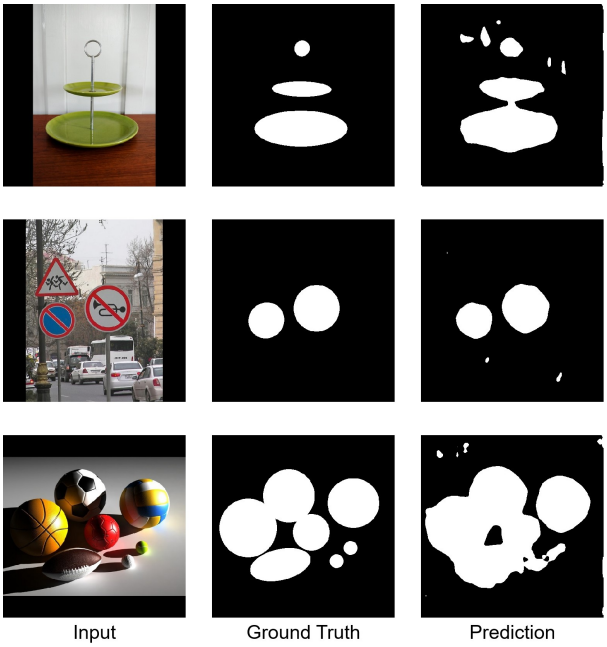


Fig. 5. Results of binary mask prediction. From left to Right are input images, ground truth and prediction of binary masks.

3.5 Ellipse Mask Segmentation

Mask segmentation [37] is able to effectively segment the boundary between the foreground objects and background. Binary masks are more sensitive to the demarcation area of the foreground and background, which can be more conducive to the utilization of elliptical edge information. Based on the idea of parameter sharing in multi-task learning, we introduce ellipse mask segmentation as an auxiliary task. Though it does not directly affect the regression of object parameters, it optimizes deep model parameters in a multi-task learning fashion and forces the model to pay higher attention to the object area. The mask loss $\mathcal{L}_M$ is calculated by binary cross-entropy loss. Some results of ellipse mask segmentation are shown in Fig 5. Overall, the total loss of our pipeline can be formulated as:

$$\begin{aligned}\mathcal{L} =&\lambda_H \mathcal{L}_H + \lambda_S \mathcal{L}_S + \lambda_O \mathcal{L}_O + \\ &\lambda_\theta \mathcal{L}_\theta + \lambda_G \mathcal{L}_G + \lambda_M \mathcal{L}_M.\end{aligned} \tag{9}$$

4 Experiments

4.1 General Ellipse Detection Dataset (GED)

To our best knowledge, most of *publicly available* ellipse datasets target for conventional methods. In particular, some samples are not regular ellipses or highly

similar. Moreover, their sizes are usually small and may not accommodate deep learning models. Thus, we contribute a new general ellipse detection dataset, termed **GED**, whose images are collected from real-world scenarios. GED consists of **1443** images and each image is manually annotated. The ellipses in GED are relatively regular and have large difference in texture. The classes of elliptic objects are diverse, *e.g.*, ball, wheel, dishes, button.

4.2 Implementation Details

For training, we use weights pre-trained on ImageNet to initialize the backbone DLA-34 [28], and Adam to optimize the networks on a single GeForce RTX 3080 GPU. For edge fusion, four edge detectors are used which are *Laplacian*, *Canny*, *Sobel* operators and *adaptiveThreshold*. All the loss weights are set as 0.1, except $\lambda_H = 1, \lambda_G = 15, \lambda_M = 1$. The aspect ratio threshold in angle loss is set to 1.2. First, the threshold of 1.2 is roughly the dividing line of the ellipse from a visual point of view. Second, approximately half of the data has the ratio of the long and short axes smaller than 1.2, while rest larger than that. Since the dataset used by EllipseNet is not open-source, we train the networks on GED and FDDB [38] dataset, respectively. 80% of the images is randomly splitted as training set and the rest as test set. All the images are resized into 512×512. The learning rate is set to be 1.25×10^{-4} and we train 150 epochs in GED dataset while 300 epochs in FDDB dataset. Traditional methods that do not use the GPU are executed on the computer with Inter Core i7-11700k 3.60 GHz.

4.3 Evaluation Metrics

We exploit two evaluation metrics: 1) AP over ellipse IoU thresholds, and 2) AP^{Θ} over ellipse IoU thresholds and angle error thresholds:

$$\begin{aligned} AP &= \sum_{i=1}^{n-1} (r_{i+1} - r_i) P_{interp}(r_{i+1}) \\ P_{interp}(r) &= \max_{r' \geq r} P(r') \end{aligned} \tag{10}$$

where $r_1, r_2, \cdots, r_n$ are the recall value. in ascending order; $P(*)$ means precision and $r(*)$ is recall, which can be computed by:

$$\begin{aligned} \text{Precision} &= \frac{TP}{TP + FP} \\ \text{Recall} &= \frac{TP}{TP + FN} \end{aligned} \tag{11}$$

We focus more on the accuracy of predicted angles by introducing AP^{Θ}. Considering the influence of round-like problem, we ignore the angle error when the aspect ration is less than 1.2. For example, $AP^{10}_{0.75}$ means that we consider a prediction as a true positive if ellipse IoU is greater than 0.75 and angle error is less than 10°, or ellipse IoU is greater than 0.75 and aspect ratio is less than 1.2.

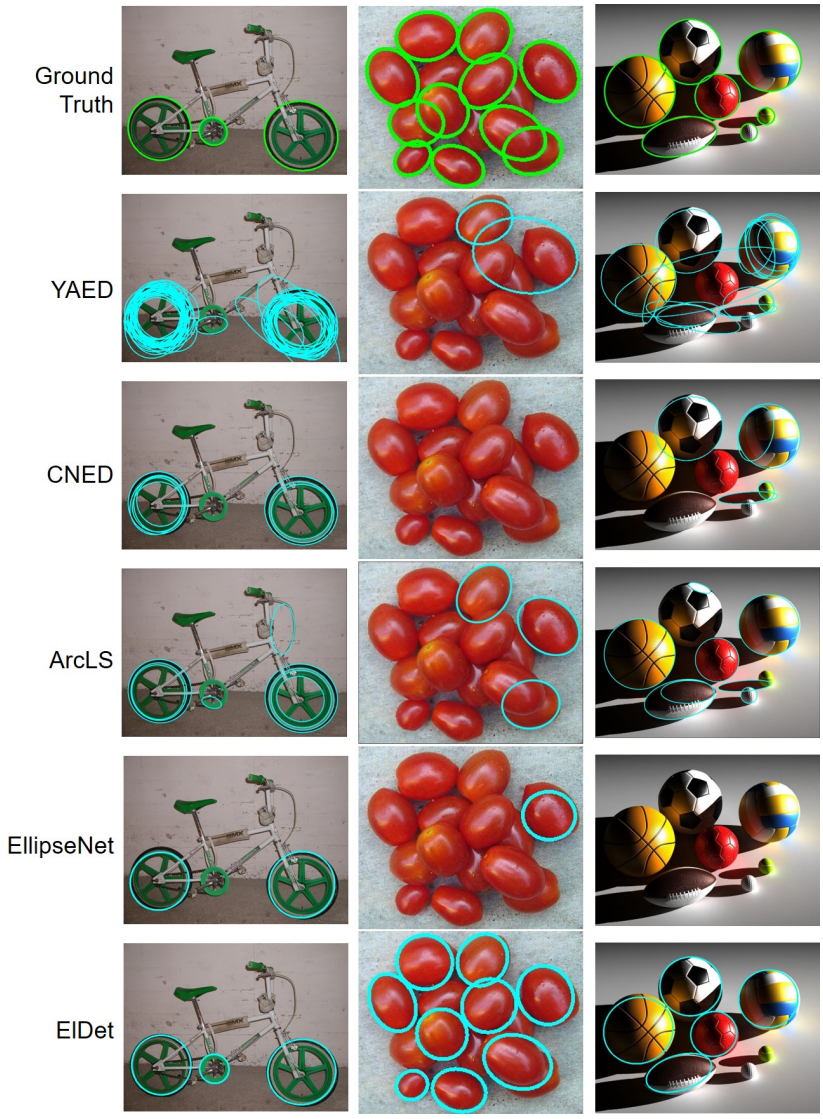


Fig. 6. Qualitative results on GED dataset. The first row shows Groud-truth images; the second to forth rows show the detections by traditional approaches; the results of fifth row is achieved by EllipseNet, which is a CNN-based approach. The last row shows visualization of our method ElDet. As we can see, our approach could successfully detect general elliptic objects.

4.4 Compared Methods

To verify the effectiveness of our *ElDet*, we compare our approach with the CNN based ellipse detection method EllipseNet [13], and also the state-of-the-art

(SOTA) traditional approaches such as YAED [39], CNED [40], and Arc-support Line Segment based method (ArcLS) [7] on GED dataset. Furthermore, we evaluate our method on FDDB dataset to explore the adaptability on downstream tasks.

4.5 Results on GED dataset.

Table 1. Quantitative results on GED dataset. $AP_{0.75}$: True positive when IoU is greater than 0.75. $AP^{10}_{0.75}$: True positive when IoU is greater than 0.75 and angle error is less than 10°.

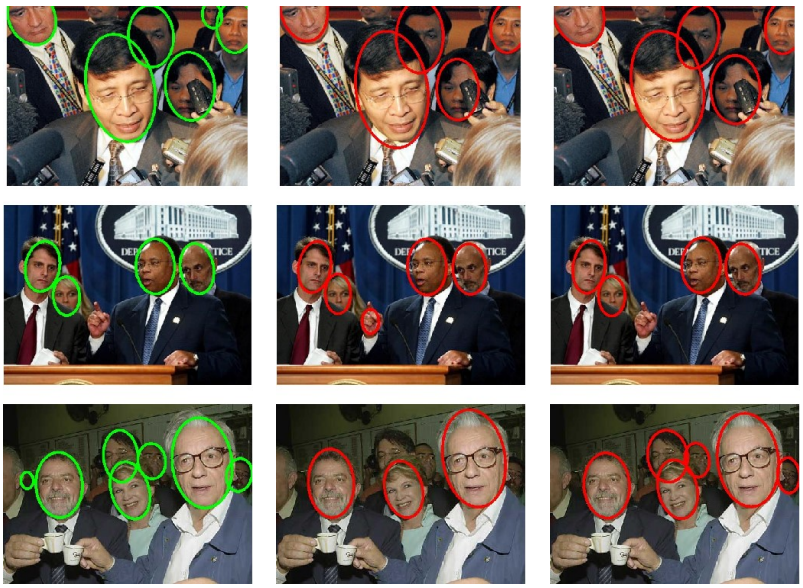
Methods	$AP_{0.5}$	$AP^{10}_{0.5}$	$AP_{0.75}$	$AP^{10}_{0.75}$	Time/ms
YAED	0.2875	0.2476	0.1942	0.1653	11.14
CNED	0.3691	0.3217	0.3134	0.2748	**9.57**
ArcLS	0.3346	0.3088	0.2312	0.2735	78.23
EllipseNet	0.3138	0.2840	0.1924	0.2735	31.96
ElDet (Ours)	**0.7403**	**0.7215**	**0.5739**	**0.5606**	44.88

Table 1 shows the detailed quantitative results. Our method achieves the best performance with a large margin compared to the state-of-the-art methods. Specifically, on $AP^{10}_{0.5}$ and $AP^{10}_{0.75}$, ElDet achieves 0.7215 and 0.5606 respectively, which indicates our model provides better accuracy of angle regression. Besides, our detection time per image is 34.94ms milliseconds on average, which is similar to EllipseNet 31.96ms. We observe that traditional approaches are highly efficient, though at lower detection accuracy compared to our ElDet. EllipseNet may be more concerned with texture information as traditional CNNs, therefore performs poorly on such a dataset with inconsistent texture information.

Moreover, we visualize some detection examples in Fig. 6 by overlaying the detected ellipses on the original image. From Fig. 6, we observe that our ElDet can handle different situations better and is robust to the interference of complex backgrounds.

Results on FDDB dataset. ArcLS regresses ellipse based on arc support line segments and requires the objects have elliptic shape. However, the faces in FDDB are not regular ellipses, thus traditional approaches are not suitable for general face detection and we only compare our ElDet with EllipseNet on FDDB dataset.

Table 2 shows the results on FDDB dataset. Our method achieves best scores with 0.8665 on FDDB dataset and outperform EllipseNet by $\sim 3\%$. The visualizations are shown in Fig. 7. As we can see, our ElDet could address some occlusions and perfectly fit face contours into ellipses. This demonstrates that our framework is rather general in addressing various geometric shapes, and has potential be adapted to various downstream tasks in real-world applications.



(a) Ground Truth (b) EllipseNet (c) ElDet

Fig. 7. Detection examples on FDDB dataset by our approach.

Table 2. Quantitative results on FDDB dataset.

Methods	$AP_{0.5}$	$AP^{10}_{0.5}$	$AP_{0.75}$	$AP^{10}_{0.75}$	Time/ms
EllipseNet	0.8383	0.6406	0.7788	0.6037	30.91
ElDet	**0.8665**	**0.8181**	**0.8218**	**0.7844**	33.77

Table 3. Quantitative results of different edge maps as data augmentation for our ElDet on GED dataset.

Methods	$AP_{0.5}$	$AP^{10}_{0.5}$	$AP_{0.75}$	$AP^{10}_{0.75}$
Adaptive Threshold	0.6599	0.6237	0.5105	0.4850
Sobel	0.7015	0.6650	0.5204	0.4954
Canny	0.7086	0.6705	0.5293	0.5148
Laplacian	0.7018	0.6802	0.5352	0.5270
Fusion	**0.7403**	**0.7215**	**0.5739**	**0.5606**

Comparison of different edge maps. Tabel 3 shows the performance of our ElDet with different edge maps as augmentation input. Among the four individual edge detection methods, *AdaptiveThreshold* method is significantly worse than other methods, and Laplacian operator obtained the best performance.

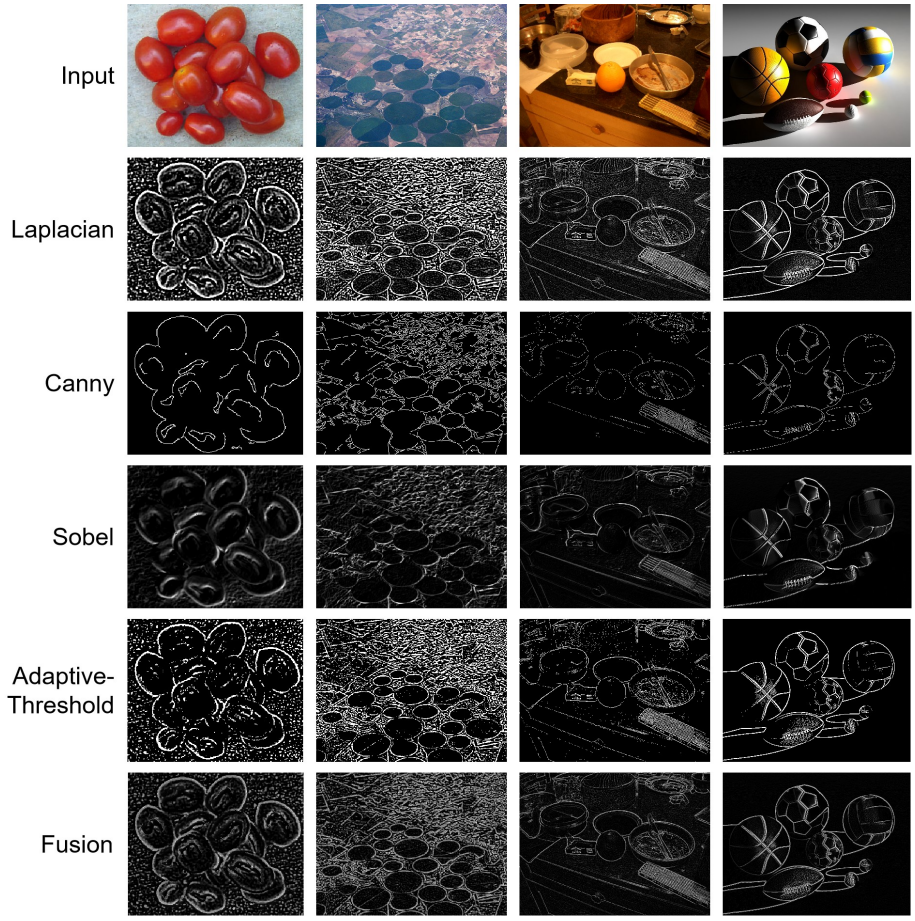


Fig. 8. Qualitative results of different edge extraction methods. From top to bottom are *Input*, *Laplacian*, *Canny*, *Sobel*, *AdaptiveThreshold* and our *Fusion* maps, respectively.

Our Edge Fusion Module (EFM) significantly outperforms individual edge map, with an improvement of about 4% in each metric. As shown in Fig 8, the noise is clearly faded and the edges are relatively clear and complete in our fused edge maps (see last row in Fig 8), which demonstrates the effectiveness of our EFM.

Ablation study. Table 4 shows the effects of proposed losses and edge information based data augmentation for ElDet. Firstly, with Gaussian IoU, the performance boosts significantly, *e.g.*, $AP_{0.75}$ increases by 0.09 and $AP^{10}_{0.75}$ increases by 0.14. It shows Gaussian IoU is suitable for object detection with rotation angle. Then, a large margin of improvement can be observed when edge information is added. Finally, when all components including angle loss weight and binary mask prediction are added, the accuracy is further improved.

Table 4. Ablation results on GED dataset. **G**: Gaussian IoU Loss. **W**: Angle Loss Weight. **M**: Ellipse Mask Segmentation. **E**: Edge Map Fusion as Data Augmentation.

Methods	G	W	M	E	$AP_{0.75}$	$AP^{10}_{0.75}$
ElDet	-	-	-	-	0.3624	0.3094
ElDet	✓	-	-	✓	0.4598	0.4439
ElDet	✓	✓	-	-	0.4378	0.4163
ElDet	✓	✓	-	✓	0.4873	0.4571
ElDet	✓	✓	✓	-	0.4513	0.4413
ElDet	✓	✓	✓	✓	**0.5739**	**0.5606**

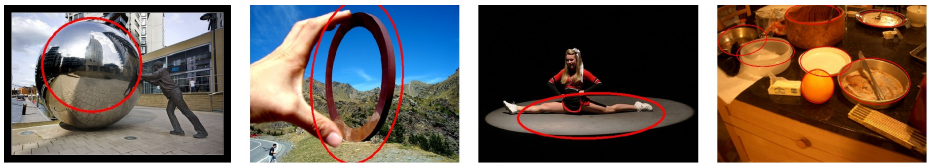


Fig. 9. Some failure cases of our ElDet method.

5 Conclusion and Discussion

In this paper, we present a simple yet anchor-free ellipse detector and contribute a general ellipse detection dataset for evaluation. Specifically, we first obtain an overall better edge map by edge fusion module which can learn the weight coefficients adaptively and take advantage of the extracted edge information as augmented input. Then, we propose angle loss weight, binary mask prediction loss, and Gaussian IoU loss to jointly improve the CNN model performance. Extensive experiments validate that our detector could provide competitive performance on both self-collected dataset and downstream task such as face detection.

Currently, we believe that our approach still has much space to improve in future research. In some scenarios, our method may fail to detect the ellipses, as shown in Fig 9. Sometimes the elliptic objects are too large or too small compared to the full image. As a consequence, the detected ellipses have larger deviation to the groundtruth, which possibly is due to the downsampling and model fitting. Moreover, for circular or elliptic objects with overlapping or occlusions, the ellipse regression accuracy may be affected under such situation. It should note that detecting overlapping objects is a common challenge for all anchor-free object detectors. For some complex scenes with low contrast, low illumination, or many ellipses, we observe that ellipse detection is still very challenging.

References

1. Lu, W., Tan, J.: Detection of incomplete ellipse in images with strong noise by iterative randomized hough transform (irht). Pattern Recognition **41**(4), 1268–1279 (2008)

2. Roy, P., Kislay, A., Plonski, P.A., Luby, J., Isler, V.: Vision-based preharvest yield mapping for apple orchards. Computers and Electronics in Agriculture **164**, 104897 (2019)
3. Lu, C., Wang, H., Gu, C., Wu, K., Guan, X.: Viewpoint estimation for workpieces with deep transfer learning from cold to hot. In: International Conference on Neural Information Processing. pp. 21–32. Springer (2018)
4. Lu, C., Gu, C., Wu, K., Xia, S., Wang, H., Guan, X.: Deep transfer neural network using hybrid representations of domain discrepancy. Neurocomputing **409**, 60–73 (2020)
5. Prasad, D.K., Leung, M.K., Cho, S.Y.: Edge curvature and convexity based ellipse detection method. Pattern Recognition **45**(9), 3204–3221 (2012)
6. Lu, C., Xia, S., Huang, W., Shao, M., Fu, Y.: Circle detection by arc-support line segments. In: 2017 IEEE International Conference on Image Processing (ICIP). pp. 76–80. IEEE (2017)
7. Lu, C., Xia, S., Shao, M., Fu, Y.: Arc-support line segments revisited: An efficient high-quality ellipse detection. IEEE Transactions on Image Processing **29**, 768–781 (2019)
8. Li, Y.: Detecting lesion bounding ellipses with gaussian proposal networks. In: International Workshop on Machine Learning in Medical Imaging. pp. 337–344. Springer (2019)
9. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28** (2015)
10. Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F.A., Brendel, W.: Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. arXiv preprint arXiv:1811.12231 (2018)
11. Kendall, A., Gal, Y., Cipolla, R.: Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7482–7491 (2018)
12. Caruana, R.: Multitask learning. Machine learning **28**(1), 41–75 (1997)
13. Chen, J., Zhang, Y., Wang, J., Zhou, X., He, Y., Zhang, T.: Ellipsenet: Anchor-free ellipse detection for automatic cardiac biometrics in fetal echocardiography. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 218–227. Springer (2021)
14. Dong, W., Roy, P., Peng, C., Isler, V.: Ellipse r-cnn: Learning to infer elliptical object from clustering and occlusion. IEEE Transactions on Image Processing **30**, 2193–2206 (2021)
15. Qian, W., Yang, X., Peng, S., Yan, J., Guo, Y.: Learning modulated loss for rotated object detection. In: Proceedings of the AAAI conference on artificial intelligence. vol. 35, pp. 2458–2466 (2021)
16. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767 (2018)
17. Huang, L., Yang, Y., Deng, Y., Yu, Y.: Densebox: Unifying landmark localization with end to end object detection. arXiv preprint arXiv:1509.04874 (2015)
18. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: A simple and strong anchor-free object detector. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020)
19. Law, H., Deng, J.: Cornernet: Detecting objects as paired keypoints. In: Proceedings of the European conference on computer vision (ECCV). pp. 734–750 (2018)
20. Lu, C., Koniusz, P.: Few-shot keypoint detection with uncertainty learning for unseen species. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 19416–19426 (June 2022)

21. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7291–7299 (2017)
22. Neubeck, A., Van Gool, L.: Efficient non-maximum suppression. In: 18th International Conference on Pattern Recognition (ICPR'06). vol. 3, pp. 850–855. IEEE (2006)
23. Xu, Y., Fu, M., Wang, Q., Wang, Y., Chen, K., Xia, G.S., Bai, X.: Gliding vertex on the horizontal bounding box for multi-oriented object detection. IEEE transactions on pattern analysis and machine intelligence **43**(4), 1452–1459 (2020)
24. Yang, X., Yan, J.: Arbitrary-oriented object detection with circular smooth label. In: European Conference on Computer Vision. pp. 677–694. Springer (2020)
25. Yang, X., Hou, L., Zhou, Y., Wang, W., Yan, J.: Dense label encoding for boundary discontinuity free rotation detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15819–15829 (2021)
26. Yang, X., Yan, J., Ming, Q., Wang, W., Zhang, X., Tian, Q.: Rethinking rotated object detection with gaussian wasserstein distance loss. In: International Conference on Machine Learning. pp. 11830–11841. PMLR (2021)
27. Yang, X., Yang, X., Yang, J., Ming, Q., Wang, W., Tian, Q., Yan, J.: Learning high-precision bounding box for rotated object detection via kullback-leibler divergence. Advances in Neural Information Processing Systems **34** (2021)
28. Yu, F., Wang, D., Shelhamer, E., Darrell, T.: Deep layer aggregation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2403–2412 (2018)
29. Zhou, X., Wang, D., Krähenbühl, P.: Objects as points. arXiv preprint arXiv:1904.07850 (2019)
30. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 2980–2988 (2017)
31. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
32. Canny, J.: A computational approach to edge detection. IEEE Transactions on pattern analysis and machine intelligence **6**, 679–698 (1986)
33. Gao, W., Zhang, X., Yang, L., Liu, H.: An improved sobel edge detection. In: 2010 3rd International conference on computer science and information technology. vol. 5, pp. 67–71. IEEE (2010)
34. Jain, R., Kasturi, R., Schunck, B.G., et al.: Machine vision, vol. 5. McGraw-hill New York (1995)
35. Bradley, D., Roth, G.: Adaptive thresholding using the integral image. Journal of graphics tools **12**(2), 13–21 (2007)
36. Panaretos, V.M., Zemel, Y.: Statistical aspects of wasserstein distances. arXiv preprint arXiv:1806.05500 (2018)
37. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)
38. Jain, V., Learned-Miller, E.: Fddb: A benchmark for face detection in unconstrained settings. Tech. rep., UMass Amherst technical report (2010)
39. Fornaciari, M., Prati, A., Cucchiara, R.: A fast and effective ellipse detector for embedded vision applications. Pattern Recognition **47**(11), 3693–3708 (2014)
40. Jia, Q., Fan, X., Luo, Z., Song, L., Qiu, T.: A fast ellipse detector using projective invariant pruning. IEEE Transactions on Image Processing **26**(8), 3665–3679 (2017)