EDGE WASSERSTEIN DISTANCE LOSS FOR ORIENTED OBJECT DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Regression loss design is an essential topic for oriented object detection. Due to the periodicity of the angle and the ambiguity of width and height definition, traditional L1-distance losses and its variants have been suffered from the metric discontinuity and the square-like problem. As a solution, the distribution based methods show significant advantages by representing oriented boxes as distributions. Differing from exploited the Gaussian distribution to get analytical form of distance measure, we propose a novel oriented regression loss, Wasserstein Distance(EWD) loss, to alleviate the square-like problem. Specifically, for the oriented box(OBox) representation, we choose a specially-designed distribution whose probability density function is only nonzero over the edges. On this basis, we develop Wasserstein distance as the measure. Besides, based on the edge representation of OBox, the EWD loss can be generalized to quadrilateral and polynomial regression scenarios. Experiments on multiple popular datasets and different detectors show the effectiveness of the proposed method.

1 INTRODUCTION

Oriented object detection has been a fundamental task in various vision area, such as aerial imagesXia et al. (2018)Liu et al. (2017b), scene textKaratzas et al. (2015)Yao et al. (2012), facesShi et al. (2018), and retail scenesChen et al. (2020), it aims to precisely locate the arbitrarily oriented objects in a single image. Although some methods based on improvements of horizontal object detection have been used in oriented object detection Yang et al. (2019)Qian et al. (2021a)Yang et al. (2021b)Yang et al. (2021c)Yang et al. (2018)Liu et al. (2017b) , it is a relatively new direction with many unsolved problems.

Designing a proper regression loss is essential for training a well-performed oriented detector. However, due to the periodicity of the angle and the ambiguity of width and height definition, the regression process has been suffering from metric discontinuity and square-like problems. Recently, some worksYang et al. (2019)Qian et al. (2021a) try to modify the L1 distance loss in order to solve these limitations. They either bring in extra regression term or depends on new representation definition, and the resultant performance is far from satisfactory. Among all the literature on oriented regression loss design, GWDYang et al. (2021b) and KLDYang et al. (2021c) are the first to propose representing the OBox as distributions. They convert the OBox to Gaussian distributions and use Wasserstein distance or Kullback-Leibler Divergence as the regression loss. The distributional representation elegantly solves the periodic problem of angle and ambiguity problem of width and height. However, the Gaussian distribution degenerates for squares and thus fails to capture the angular information as showed in Figure-1(b).Besides, those near square objects will also have trouble in regressing the angle precisely(shown in Figure-2) as their optimization process is near stopping. Therefore, we believe that there should be a much more reasonable choice other than the Gaussian representation to fully exploit the advantage of the distributional representation method. In this paper, we follow the idea of distributional representation and propose a novel oriented regression loss called Edge Wasserstein Distance(EWD) loss to solve all the above problems.

In order to better reserve the geometrical information of OBox, we introduce a specially-designed distribution whose probability density function is only nonzero over the edges as its representation. In this way, we develop the Wasserstein distance as the loss to regress the distribution between the ground-truth and prediction. As it is difficult to compute this distribution loss directly, we

(a) 5-p representation   (b) Gaussian representation   (c) Edge representation
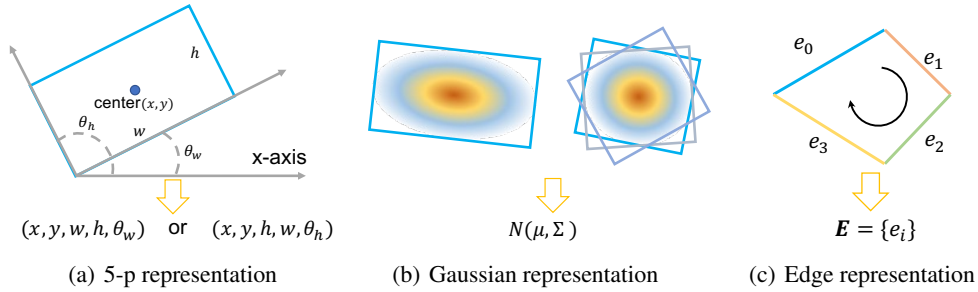
Figure 1: Illustration of oriented bounding box representations. a) Using five parameters (x, y, w, h, θ) to represent a rotated bounding box. A rotated box can be mapped to two variants in the 5-p vector space. b) Gaussian representation proposed by GWDYang et al. (2021b). It converts the rotated box to Gaussian distribution. However, multiple squares are likely to be mapped to one Gaussian. c) Our proposed edge representation. We use a clockwise(or counter-clockwise) edge sequence to represent the rotated box. It is not affected by the square-like problem.

introduce several assumptions to simplify the calculation and present a simple approximation of Edges Wasserstein distance(EWD) as the regression loss. The proposed EWD has several advantages. First, with the distributional representation, it solves all the limitations of previous work, immune from both the angular periodicity and ambiguity of width and height. Also the square-like problem will not exist. Second, by treating the OBox as a whole, the optimization of boxes' parameters are dynamically adjusted according to the chosen target. This leads to high performance especially in high-precision cases. Third, the proposed EWD, along with the edge representation, is more general compared to other distribution based methods. It is applicable to quadrilateral and polynomial regression scenarios. In summary, the contributions of this paper are three folds:

1) We introduce edge representation and propose a novel oriented regression loss, Edge Wasserstein Distance loss, which is simple and general to rectangular, quadrilateral and polynomial regressions.

2) From general to specific, we theoretically develop the formulation of EWD loss on oriented bounding box occasions. The resultant formulation is simple and turns out to be a generalized form of horizontal L2-distance in oriented cases.

3) Experimentally, we show that for three datasets and two popular detectors, the EWD loss can achieve challenging results compared with its peer methods.

2 BACKGROUND

2.1 RELATED WORK

Oriented object detection is an extension to classical horizontal detection. It is widely used in areas like aerial object detectionHe et al. (2016), scene text detectionZhou et al. (2017)Long et al. (2018)Liao et al. (2020) where targets need to be more precisely located. To find the orientation, there are mainly three approaches: the classification based Yang et al. (2021a)Yang & Yan (2020), segmentation basedWang et al. (2019)Liao et al. (2020) and regression based methodsZhou et al. (2017)Liu et al. (2017a)Jiang et al. (2017)Qian et al. (2021b). The latter approach is preferred by most researchers for its simplicity and advantage in getting high-precise localization. However, the oriented regression has been suffered from discontinuity problem and square-like problem when using the traditional L1 loss. These two problems are mainly caused by the periodicity of angle and variations in bounding box definition. As a remedy for L1 loss, SCRDetYang et al. (2019) proposes IoU-Smooth L1 loss to smooth the boundary of discontinuity by introducing the skew-IoU factor. RSDetQian et al. (2021a) proposes modulated loss to overcome the discontinuity. Although they partly solve these problems, the performances are far from satisfactory. Other methods try to introduce the IoU loss Yu et al. (2016)Zheng et al. (2020)Rezatofighi et al. (2019) in horizontal detection into oriented detection. They usually approximate the IoU loss as its implementation in oriented occasions is too complicated. PolarMaskXie et al. (2021) proposes an approximation of IoU operation. However, it is discrete and cannot be directly used for back propagation. Similar idea is taken by PIoUChen et al. (2020). The most similar work to our method is GWDYang et al.

(2021b) and KLDYang et al. (2021c), which both represent the oriented bounding box as Gaussian distribution. However, the square-like problem still exists. Moreover, they are limited to regressing the rectangular boxes and can not be generalized to quadrilateral cases.

2.2 Oriented Bounding Box Representation: A Distributional Perspective.

For representing a rotated bounding box, the most commonly used method is the 5-p representation as denoted in Figure-1(a). As has been mentioned, when it is used together with L1 loss, the problem of metric discontinuity and square-like problems will arise. To avoid the limitations of 5-p representation and to make the regression more accurate, GWD and KLD are first proposed to represent the rotated box as Gaussian distribution (μ, Σ), where $\mu = (x, y)$, $\Sigma^{1/2} = RSR^\top$ and R is the rotational matrix defined by θ. It solves the problem of ambiguity between width and height. On this basis, the boxes' distance can be calculated via popular distributional distance measure, KL-Divergence or Wasserstein distance. The distributional representation further promotes high-precision performance by treating the the bounding box as a whole. In such way, the optimization process of each parameter is adaptively adjusted. By comparison, the 5-p representation together with L1 loss treat each parameter independently, thus getting sub-optimal performance.

Following the distributional representation, we find that Gaussian may not be a perfect choice. As denoted in Figure-1(b), the Gaussian representation degenerates for the squared OBoxs as the rotation of squares will not change their Gaussian representation. In other words, Gaussian representation is not a one-to-one mapping. It loses the angular information for squares.

To better represent the OBox, we design a distribution to capture the full geometrical information, which we call edge representation(shown in Figure-1(c)). Specifically, we use a directional edge sequence $\mathbf{E}$ to represent a rotated box and we design a 2-dimensional distribution which is only defined on the edges and zero-valued elsewhere. The probability density function of OBox is written as

$$p(x, y) = \begin{cases} \tilde{p}(x, y) & (x, y) \text{ on } \mathbf{E} \\ 0 & \text{Otherwise} \end{cases} \tag{1}$$

where $\tilde{p}$ satisfies $\int_{\mathbf{E}} \tilde{p}(x, y) \, ds = 1$. The edge representation is general and can be applied to quadrilateral and polynomial cases. Next, we will develop the distance measure based on the edge representation.

3 Proposed Method

3.1 Edge Wasserstein Distance

Based on the previous edge representation, we formulate the distance between two OBoxes' distributions, $\mathbf{p}, \mathbf{q}$ by the Wasserstein metric

$$\mathbf{W}(\mathbf{p}, \mathbf{q}) = \inf \mathbb{E}(||\mathbf{X} - \mathbf{Y}||_\mathbf{2}^\mathbf{2}) \tag{2}$$

where $\mathbf{X}$ and $\mathbf{Y}$ are points sampled from $\mathbf{p}$ and $\mathbf{q}$. However, as the definition of Wasserstein distance involves an optimization process, the calculation of Eq-2 is nontrivial. As a simplification, we make an assumption on the joint distribution of two sampled points $\mathbf{X}$ and $\mathbf{Y}$. Suppose $\mathbf{P_i}$ and $\mathbf{Q_i}$ are corner points of two OBoxes where $i \in \{0, 1, 2, 3\}$ and $\mathbf{X_i}$ and $\mathbf{Y_j}$ are sampled from segment $\mathbf{P_i P_{i+1}}$ and $\mathbf{Q_j Q_{j+1}}$ separately. We restrict $p(\mathbf{X}, \mathbf{Y})$ by

$$p(\mathbf{X_i}, \mathbf{Y_j}) = \begin{cases} \tilde{p}(\mathbf{X_i}, \mathbf{Y_j}) & j = \pi(\mathbf{i}) \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where π denotes a bipartite matching function which maps the edge index i from one OBox to $\pi(\mathbf{i})$ of the other. The assumption restricts the distance calculation only between the geometrical edge counterparts. Thus, the 2 can be simplified as

$$\begin{aligned} \mathbf{W}(\mathbf{p}, \mathbf{q}) &= \inf_{p(\mathbf{X_i}, \mathbf{Y_j})} \mathbb{E} \sum_i \sum_j (||\mathbf{X_i} - \mathbf{Y_j}||_\mathbf{2}^\mathbf{2}) \\ &= \inf_\pi \sum_i \Big(\inf_{p(\mathbf{X_i}, \mathbf{Y_{\pi(i)}})} \mathbb{E} ||\mathbf{X_i} - \mathbf{Y_{\pi(i)}}||_\mathbf{2}^\mathbf{2} \Big) \end{aligned} \tag{4}$$

The distance calculation of the OBoxes is divided into the distance measure of each segment pair $\mathbb{E}||\mathbf{X_i} - \mathbf{Y_{\pi(i)}}||_2^2$. For the minimum operation of the bipartite matching function π, as the direction of the edge sequence has been defined, the number of total matched pairs is equal to the number of edges. In practice, we traverse all the matches and get the minimum. Further simplification of Eq-4 depends on either the distribution $\tilde{p}(x, y)$ of the OBox edge or the joint distribution $\tilde{p}(\mathbf{X_i}, \mathbf{Y_j})$. Here, we introduce two approximations to simplify the Eq-4 and formulate the Edge Gaussian Wasserstein Distance(EGWD) and Edge Dense Wasserstein Distance(EDWD).

Edge Gaussian Wasserstein Distance (EGWD) We suppose $\tilde{p}(x, y)$ to be a 2-D Gaussian distribution denoted by (μ, Σ), where μ represents the edge's center point and Σ is the covariance. The covariance is calculated as $\Sigma^{1/2} = \mathbf{R}\mathbf{S}\mathbf{R}^\top$ where $\mathbf{R}$ is the rotation matrix and $\mathbf{S} = diag(\frac{w}{2}, 0)$ with w being the edge length. Following the above derivation, it turns out that the Wasserstein distance between two consistent edge pair(μ_1, Σ_1) and (μ_2, Σ_2) is written asChafaï (2010):

$$\mathbf{W_{12}} = ||\mu_1 - \mu_2||_2^2 + \mathbf{Tr}\left(\Sigma_1 + \Sigma_2 - 2(\Sigma_1^{1/2}\Sigma_2\Sigma_1^{1/2})^{1/2}\right) \tag{5}$$

Thus, the Eq-4 can be written as the summation of each paired distance.

Edge Dense Wasserstein Distance (EDWD) In this part, we further restrict the joint distribution for $\tilde{p}(\mathbf{X_i}, \mathbf{Y_j})$ by Eq-6. In this way, the optimization problem is avoided and the calculated distance becomes an upper bound for the exact Wasserstein distance.

$$p(\mathbf{X_i}, \mathbf{Y_j}) = \begin{cases} \tilde{p}(\mathbf{X_i}, \mathbf{Y_j})) & j = \pi(\mathbf{i}) \text{ and } \delta(\mathbf{X_i}) = \delta(\mathbf{Y_i}) \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $\delta(\mathbf{X_i}) = ||\mathbf{X_i}\mathbf{P_i}||/||\mathbf{P_i}\mathbf{P_{i+1}}||$ representing the normalized position of $\mathbf{X_i}$ on the segment. It means that each point on the edge is matched exactly to its counterparts on the paired edge. As the sampling process of $\mathbf{X_i}$ and $\mathbf{Y_i}$ becomes fully coupled, the joint distribution $\tilde{p}(\mathbf{X_i}, \mathbf{Y_j})$ degenerates to the edge's distribution, which is now a one dimensional distribution. Then Eq-4 can be further simplified by a line integration. We directly show the final formulation for Eq-2 and leave the derivation to the appendix.

$$\mathbf{W}(\mathbf{p}, \mathbf{q}) = \inf_\pi \sum_\mathbf{i} \left(||\mathbf{\Delta c_i}||^2 + \frac{\sigma^2}{4}||\mathbf{\Delta w_i}||^2\right) \tag{7}$$

where $\mathbf{c_i}$ denotes center of the segment, $\mathbf{\Delta c_i}$ denotes the vector between paired segments' center points, $\mathbf{w_i}$ denotes the edge vector and $\mathbf{\Delta w_i}$ denotes the vector difference. σ^2 denotes the variance of the designed paired edges' distribution $\tilde{p}$. The geometrical meaning of this approximation is to force every point sampled from the edge to be densely aligned to its counterparts. Hence, the regression can be viewed as a dense alignment process.

3.2 Edge Wasserstein Distance Regression Loss for Oriented Bounding Box

The developed EGWD and EDWD in Eq-12 and Eq-28 are for general polygons. For OBoxes, the edges are not independent. Thus, we can further bring in the geometrical constraint and develop EWD for OBox. Without loss of generality, we take the θ-based representation (x, y, w, h, θ) for OBox.

EGWD for OBox. There are two constraints on the condition of OBox. First, the center points of each edge can be expressed by the 5-p vector (x, y, w, h, θ). Second, the parallel edges share identical covariance. By clockwise order, the center points can be written as

$$\mu_1 = (x - \frac{h}{2}\sin\theta, y - \frac{h}{2}\cos\theta)^\top, \mu_2 = (x + \frac{w}{2}\cos\theta, y - \frac{w}{2}\sin\theta)^\top$$
$$\mu_3 = (x + \frac{h}{2}\sin\theta, y + \frac{h}{2}\cos\theta)^\top, \mu_4 = (x - \frac{h}{2}\sin\theta, y + \frac{w}{2}\cos\theta)^\top \tag{8}$$

And the two covariance matrices that correspond to w and h are written as:

$$\Sigma_w^{1/2} = \begin{pmatrix} \frac{w}{2}\cos^2\theta & \frac{w}{2}\cos\theta\sin\theta \\ \frac{w}{2}\cos\theta\sin\theta & \frac{w}{2}\sin^2\theta \end{pmatrix}, \Sigma_h^{1/2} = \begin{pmatrix} \frac{h}{2}\sin^2\theta & \frac{h}{2}\cos\theta\sin\theta \\ \frac{h}{2}\cos\theta\sin\theta & \frac{h}{2}\cos^2\theta \end{pmatrix} \tag{9}$$

Then for a given match π, the summation of the former term of Eq-5 can be simplified into

$$\sum_i ||\mu_1 - \mu_2||_2^2 = 4||\mathbf{\Delta o}||^2 + \frac{1}{2}||\mathbf{\Delta w}||^2 + \frac{1}{2}||\mathbf{\Delta h}||^2 \tag{10}$$

where $\mathbf{\Delta o} = (\mathbf{\Delta x})^2 + (\mathbf{\Delta y})^2$ represents the center point distance and $\mathbf{\Delta w} = ||(\mathbf{w_1}\cos\theta_1, \mathbf{w_1}\sin\theta_1)^\top - (\mathbf{w_2}\cos\theta_2, \mathbf{w_2}\sin\theta_2)^\top||^2$ represents the width vector difference. $\mathbf{\Delta h}$ is similar to $\mathbf{\Delta w}$. Similarly, we develop the simplified expression for the summation of latter term of Eq-5 as

$$\mathbf{Tr}(...) = \frac{1}{2}||\mathbf{\Delta w}||^2 + \frac{1}{2}||\mathbf{\Delta h}||^2 \tag{11}$$

Thus, we combine Eq-10 and Eq-11 together and get the formulation for EGWD distance measure as

$$\mathbf{W}(\mathbf{p}, \mathbf{q}) = 4||\mathbf{\Delta o}||^2 + ||\mathbf{\Delta w}||^2 + ||\mathbf{\Delta h}||^2 \tag{12}$$

The full derivation is left to the **appendix**. At first glance, we find that the formulation of EGWD is the generalized form of L2-distance loss.

EDWD for OBox. The simplification process of EDWD is similar to EGWD as they both contain the summation of edge center difference. On the basis of Eq-10, the Eq-7 can be written as

$$\mathbf{W}(\mathbf{p}, \mathbf{q}) = 4||\mathbf{\Delta o}||^2 + \left(\frac{1}{2} + \frac{\sigma_\mathbf{w}^2}{2}\right)||\mathbf{\Delta w}||^2 + \left(\frac{1}{2} + \frac{\sigma_\mathbf{h}^2}{2}\right)||\mathbf{\Delta h}||^2 \tag{13}$$

where the σ_w^2 and σ_h^2 represent the variances of the single-variate distribution that correspond to the width and the height. Their values describe the geometrical information of the corresponding edges. In this work, we design the variance to be w for width and h for the height. In this way, the variance ratio keeps the OBoxes' aspect ratio. This special design leads to better performance compared to directly setting them to a constant value. We will discuss it in the next section.

3.3 OVERALL LOSS FUNCTION DESIGN

Relationship between EGWD and EDWD. EGWD and EDWD have similar formulations. Comparing Eq-12 and Eq-28, we find that the mathematical form of EGWD can be viewed as a special case of EDWD when setting identical variance σ_w^2, σ_h^2 for each side, although they are developed under different assumptions. For EDWD, the variance is set according to the length of the edge. We will show that this property is beneficial to the optimization process in the next part.

Scale Normalization. In object detection, a good regression loss should consider the problem brought by bounding box's scales. For L1 loss, the large target will produce much larger penalty than small targets, which results in the degeneration of performance of small objects. To address this issue, we normalize both the predicted and target bounding boxes by a certain scale to make the EWD distance more robust in various scales. Specifically, we normalize the target width and height (w_t, h_t) to $(1, 1)$ and use them to scale the predicted (w_p, h_p) to $(\frac{w_p}{w_t}, \frac{h_p}{h_t})$. As for the center offset, using either w_t or h_t as the denominator is ambiguous, thus we use $\sqrt{w_t h_t}$ to scale the offset. We also use this value to scale the variance of EDWD. Then for gradient with respect to center $o_p = (x_p, y_p)$

$$\frac{\partial f_{EWD}(o_p)}{\partial o_p} = \left(\frac{2}{s^2}\Delta x, \frac{2}{s^2}\Delta y\right) \tag{14}$$

The scale factor s will dynamically adjust the gradient for various scaled boxes. Small targets will get large gradients which is important as that small deviation will produce much more severe IoU decrease for small scale targets. Similar analysis can be done for w_p and h_p.

$$\frac{\partial f_{EWD}(w_p)}{\partial w_p} = (1 + \frac{w_t}{h_t})(\frac{w_p}{w_t} - \cos\Delta\theta), \frac{\partial f_{EWD}(h_p)}{\partial h_p} = (1 + \frac{h_t}{w_t})(\frac{h_p}{h_t} - \cos\Delta\theta) \tag{15}$$

We can see that the gradient of w_p and h_p are also adjusted by the normalization scale. Smaller height or width will receive larger gradient. Besides, the gradient is also affected by the θ difference and aspect ratio. The larger side will get slightly larger gradient compared to smaller side. This is desirable especially for extremely smaller object where regression is more likely to be affected by noise. For the gradient of $\Delta\theta$, we have

$$\frac{\partial f_{EWD}(\Delta\theta)}{\partial\cos\Delta\theta} = -\left(\frac{w_p}{w_t} + \frac{h_p}{h_t} + \frac{w_p}{h_t} + \frac{h_p}{w_t}\right) \tag{16}$$
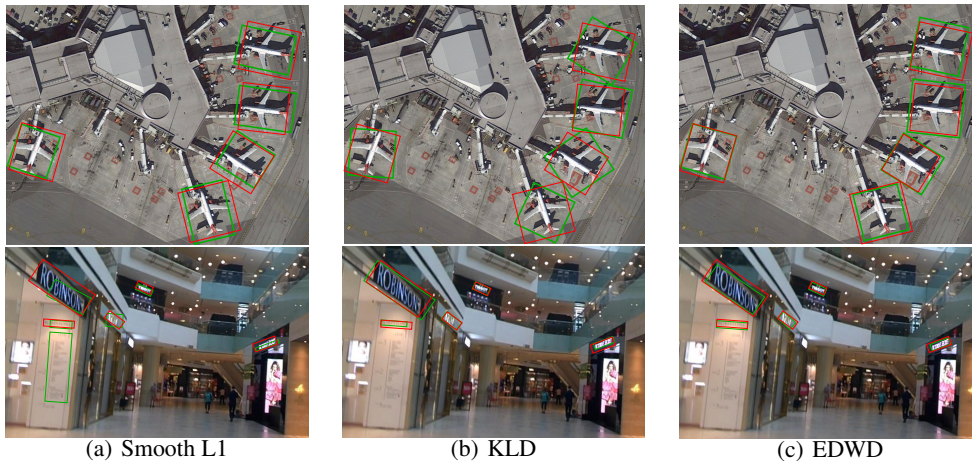
| (a) Smooth L1 | (b) KLD | (c) EDWD |

Figure 2: Visual comparison between different losses. Ground-truth bboxes are in red, predicted bboxes are in green.
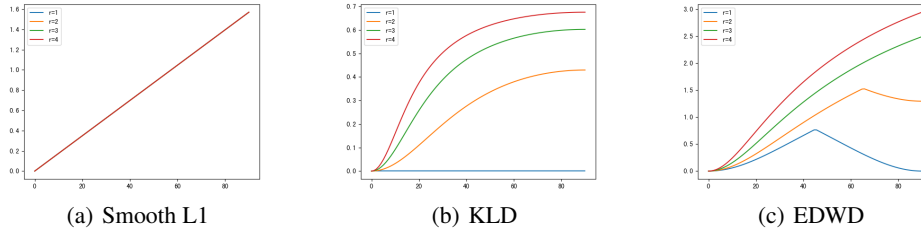

| (a) Smooth L1 | (b) KLD | (c) EDWD |

Figure 3: Smooth L1, KLD and EDWD versus angle when aspect ratio varies.

When $w_p = w_t$ and $h_p = h_t$, then $\frac{\partial f_{EWD}(\Delta\theta)}{\partial \cos\Delta\theta} = -(2 + \frac{w_t}{h_t} + \frac{h_t}{w_t}) \geq -4$, the condition for equality is that $h_t = w_t$. That is the square case. So this makes the model to produce large gradient for angle when the aspect ratio is large. For the square case, there are still gradient for the angle optimization. Remind that both KLD and GWD suffers from problem of gradient vanishing in square cases.

Post Function. We find that directly applying the EWD distance as the regression loss is unstable. In most cases, the regression process failed to converge. This is caused by the squared operation which is sensitive to large errors. To fix it, we simply apply a squared root operation to the EWD distance. Note that GWD and KLD also encounter this problem and they use a non-linear mapping function $\log(1 + x)$ or $\frac{1}{\tau + x}$ to modulate the distance. We also tried this techniques and get similar results with the squared root operation.

4 EXPERIMENTS

4.1 DATASETS AND IMPLEMENTATION DETAILS

We evaluate our method on multiple datasets mainly covering aerial images: DOTAXia et al. (2018), HRSC2016Liu et al. (2017b), and text detection: ICDAR2015Karatzas et al. (2015), MSRA-TD500Yao et al. (2012). **DOTA**Xia et al. (2018) is one of the largest data set for multi-class object detection in aerial images. It contains 15 categories, 2,806 images and 188,282 instances. Images' scales range from 800×800 to 4000×4000 pixels. In the dataset, each object is annotated by an oriented bounding box (OBox), which can be denoted as $(x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4)$, where (x_i, y_i) denotes the i-th vertice of OBox. The dataset is split into training, validation, and test sets with 1/2, 1/6, and 1/3 ratio, respectively. For the short names in Table-2, they are defined as(abbreviation-full name): BR-Bridge, SV-Small vehicle, LV-Large vehicle, SH-Ship, HA-Harbor, ST-Storage tank, RA-Roundabout, PL-Plane, BD-Baseball diamond. **HRSC2016**Liu et al. (2017b) is another aerial images data set for ship detection. It contains images from two scenarios including ships on sea and

ships close inshore. HRSC2016 dataset has four ship categories, 1061 images and 2976 samples in total. It split into training, validation and test set which contains 436 images including 1207 samples, 181 images including 541 samples and 444 images respectively. The image sizes range from 300×300, to 1500×900. **ICDAR2015**Karatzas et al. (2015) and **MSRA-TD500** are both commonly used for oriented scene text detection and spotting. ICDAR2015 includes 1,000 training images and 500 testing images. MSRA-TD500 consists of 300 training images and 200 testing images. We perform experiments on these scene text detection datasets in order to verify the scenario generality of our work.

We mainly choose two types of detectors, RetinanetLin et al. (2017b) and FCOSTian et al. (2020) to verify our methods. The implementation of the rotated version of them are based on AlpharotateYang et al. (2021d) and MMRotateZhou et al. (2022). These two detectors are specially chosen as the representation of anchor-based methods and anchor free methods. We use ResNet-50He et al. (2016) backbone with FPNLin et al. (2017a) head for all datasets. Training is conducted on single Tesla P100 GPU with 16GB memory, each with batch size of 2. For each datasets, Stochastic gradient descent(SGD) is adopted as the optimizer and the base learning rate is set to 0.0025, weight decay set to 0.0001, momentum set to 0.9. The training details for each dataset is somewhat different. For DOTA, we use the training split to train the detector 20 epochs and use the validation split to report the final results. For HRSC, ICDAR2015 and MSRA-TD500, as the these datasets are relatively small, we set the training step of one epoch to 2190, 5000 and 1500 and total epochs to 12 to ensure that the detector is fully trained. The base augmentation is simply set as random flip. If other augmentation, eg, random rotate, is applied, the training length will be increased accordingly.

4.2 COMPARISONS AND ABLATIONS

Post functions. Table-3 compares the performance of different post functions and hyper parameters. We report the results on HRSC2016 with Retinanet. Other datasets and detectors will result in similar conclusions. The results show that the EWD without post function gets poor performance. Thus, a simple $\sqrt{x}$ function will improve the performance by a large margin. Using the $\log(1+x)$ as the post function gets the best performance 89.49%. For the post function $1 - \frac{1}{\tau+x}$ proposed by GWDYang et al. (2021b), we find that the hyper parameter τ hardly affects the performance. In most of the experiments, we use $\log(1+x)$ as the post function unless explicitly specified.

Scale normalization. Table-1 shows the results with different types of scale normalization. The results are reported on DOTA-1.0 with Retinanet. We also investigate into the effect of scale normalization on smooth L1 loss. The results show that the smooth L1 loss without any normalization gets the worst results, while a simple normalization by either the image size or the anchor size will largely boost the performance(from 45.3 to 63.1 or 67.2). In practice, we find that the L1 loss without any normalization will get a large loss value, which will overwhelm the classification loss and thus deteriorate the training. So normalization by the image size or the anchor size will fix this problem by means of lowering down the influence of regression loss. Further more, by adopting the scale normalization by the target box value h_t/w_t or the minimum/maximum of them, the performance can still be improved. Normalizing by the target minimum or maximum gets the best results, both for smooth L1 loss and EWD loss. In all cases, the proposed EWD loss beats the smooth l1 loss. Even without normalization, EWD loss gets 67.2% thanks to the modulation effect of post functions.

Bounding Box Definition. The bounding box definition refers to the range restriction techniques for L1 loss(referring to Sec-2.2). In Table-2, we show the results of smooth L1 loss with different bounding box definitions. We find that the minimum angle definition(restricting the angle to be in $[-45°, 45°)$ shows large advantage over the other two definitions. However, in other two scene text datasets, we find that the long edge definition usually works best, which is chosen as the baseline.

Square-like problem. In Table-2, we compare the performance on circular targets:storage tank(ST), roundabout(RA) and square-like targets:plane(PL), baseball diamond(BD). The circular target is a special kind of square-like target whose angular direction is not defined. It is shown that EWD loss gets slightly higher results than KLD and GWD for circular objects. They all behave much better than the smooth L1 loss and its variants. For square-like objects, we report the AP_{85} on PL and BD. The AP_{50} is not chosen because of squares with only angle difference will always get IoU higher than 82%. It is shown that the KLD and GWD methods behaves even worse than Smooth L1 loss while our method performs the best. Moreover, we visualize the OBox predicted by L1 loss, KLD methods

and ours. It is shown in Figure-2. For KLD loss, it fails to predict the direction of the square object. But the L1 loss and ours do not suffer from it. The square-like problem can also be understood by the distance-angle curve shown in Figure-3. We plot the distance compared by L1-loss, KLD and EWD on different aspect ratio objects. For square-like objects(aspect r=1, the blue curve), the distance of KLD is always zero. While L1 and EWD do not have this problem.

Table 1: Ablations of scale normalization on DOTA-1.0. w/o and w/ are short for without and with.

method	loss	w/o norm	w/ norm by				
			image size	anchor size	h_t / w_t	$\min(h_t, w_t)$	$\max(h_t, w_t)$
Retinanet	Smooth-L1	45.3	63.1	67.2	67.8	68.2	68.1
	EWD	67.2	66.9	68.1	68.8	**69.6**	69.4

Table 2: Comparisons on DOTA-1.0. Refer to Sec-4.1 for short names. The symbols, †, ○, □, represent large aspect ratio, circular and near-square objects respectively. The bold red and blue fonts indicate the top two performance. The oc, le and min denote the opencv definition($\theta \in [-90°, 0°)$) , long edge definition and minimum theta definition($\theta \in [-45°, 45°)$ of oriented bounding box.

detector	method	Box Def	$BR^\dagger$	$SV^\dagger$	$LV^\dagger$	$SH^\dagger$	$HA^\dagger$	$ST°$	$RA°$	$PL^\square_{85}$	$BD^\square_{85}$	5-AP_{50}	AP_{50}	AP_{75}	$AP_{50:95}$
Retinanet	Smooth L1	oc	34.34	55.84	66.88	75.16	55.51	83.42	59.03	27.37	6.85	57.54	63.42	35.95	36.07
		le	36.97	63.84	70.99	82.80	59.33	82.70	64.94	24.96	0.91	62.78	65.41	36.14	37.17
		min	35.21	65.18	73.02	85.25	58.07	82.23	62.33	40.97	6.13	63.34	67.26	38.29	39.00
	IoU-Smooth L1 Yang et al. (2019)	min	36.44	62.25	74.01	86.43	58.17	81.08	63.01	39.71	6.08	63.46	67.19	38.21	39.14
	Modulated Loss Qian et al. (2021a)	-	36.30	64.01	69.33	82.76	61.20	82.53	61.29	35.15	5.73	62.72	66.31	37.32	38.61
	GWD Yang et al. (2021b)	-	**41.03**	66.30	73.80	83.31	61.76	83.82	63.33	31.03	2.89	65.24	68.61	40.68	40.71
	KLD Yang et al. (2021c)	-	40.97	**67.21**	**76.52**	**86.20**	**63.10**	**84.08**	66.01	31.99	3.29	**66.8**	68.14	**44.48**	**42.15**
	EGWD	-	36.26	66.16	72.15	85.54	61.20	84.02	**66.20**	33.70	**6.15**	64.26	**69.10**	42.62	40.73
	EDWD	-	39.57	**66.95**	**76.29**	**87.01**	**62.01**	**84.25**	**66.45**	**42.20**	**7.12**	**66.36**	**69.61**	**43.60**	**41.66**
FCOS	Smooth L1-Loss	min	39.60	67.70	72.91	85.04	60.65	83.03	67.40	22.07	9.40	65.18	68.42	29.46	35.30
	GWD Yang et al. (2021b)	-	**42.73**	68.12	**82.48**	**87.86**	65.90	83.67	67.31	27.03	9.85	69.62	**70.00**	38.20	39.70
	KLD Yang et al. (2021c)	-	**43.52**	**69.70**	81.67	87.41	65.48	84.82	**67.87**	28.46	8.00	**69.56**	69.94	39.52	**40.93**
	EGWD	-	41.79	68.12	81.79	87.21	**67.31**	**84.95**	67.80	**31.16**	**10.86**	69.24	69.83	**41.38**	40.26
	EDWD	-	42.01	**68.52**	**82.78**	**88.00**	**70.73**	**85.41**	**68.82**	**38.04**	**11.04**	**70.41**	**70.91**	**41.45**	**41.51**

Table 3: Ablations on post functions. The results are AP_{50} reported on HRSC2016 Liu et al. (2017b).

	$f(W)$	$1 - \frac{1}{\tau + f(W)}$			
		$\tau = 1$	$\tau = 2$	$\tau = 3$	$\tau = 5$
$f(W) = W$	60.01	88.78	88.91	88.66	88.51
$f(W) = \sqrt{W}$	88.27	88.45	87.90	88.21	88.35
$f(W) = \log(1 + W)$	**89.49**	89.21	88.13	87.72	88.21

Table 4: Results of quadrilateral regression on HRSC2016.

method	Smooth-L1	Modulated Loss Qian et al. (2021a)	Gliding Vertex Xu et al. (2020)	EDWD
AP50	72.06	88.51	87.23	**90.01**
mAP50:95	35.13	53.07	50.33	**59.62**

Comparison with peer methods. In this part, we compare EWD loss with smooth L1 loss, IoU-Smooth L1 Loss Yang et al. (2019), Modulated loss Qian et al. (2021a), GWD Yang et al. (2021b) loss and KLD Yang et al. (2021c) loss. These methods are chosen as they are all designed to solve the problems for oriented regression. GWD and KLD are most relevant to our method and the comparison with them is conducted on all datasets and settings. The comparison is conducted on three commonly used datasets, DOTA-1.0(Table-2), HRSC2016(Table-5 and ICDAR2015(Table-6. We choose two most representative detectors, Retinanet and FCOS, and implement their rotated version as the base detectors. The rotated FCOS increases overall baseline performance compared to rotated Retinanet, but it gets a worse result on high-IoU occasions. Nevertheless, we can see that on all occasions, the EDWD loss gets the best results. For example, using rotated Retinanet on DOTA-1.0, the EDWD loss increased the AP_{50} by 2.35% compared to the best performed Smooth L1 loss. It also beats the KLD loss by 1.47%.

Large aspect ratio optimization. The EWD loss and KLD are both optimized for large aspect ratio targets. Mathematically, for fixed angle deviation, larger aspect ratio targets result in much larger penalty, as shown in Figure-3. While it makes no difference for L1 loss. It is noticed that there is a sharp turning point on the loss curve of EWD loss for small aspect ratio objects. As long as the aspect ratio is larger than a certain limit, the turn point vanishes. The turning point is caused by the infimum operation in Eq-7. It means that when the angle difference is larger than the boundary value, the current bipartite match has to change in order to get the minimum. In Table-2, we choose

Table 5: Experiments on HRSCLiu et al. (2017b).

Detector	Methods	Smooth L1	GWD	KLD	EGWD	EDWD
Retinanet	AP50	84.28	85.56	87.45	88.87	**89.49**
	AP50:95	47.76	52.89	58.72	56.19	**58.80**
FCOS	AP50	88.30	89.74	89.85	89.70	**90.30**
	AP50:95	51.82	54.12	55.03	53.98	**55.32**

Table 6: Comparisons on ICDAR2015

Detector	Method	No Rotate Aug			Rotate Aug		
		Hmean50	Hmean75	Hmean50:95	Hmean50	Hmean75	Hmean50:95
Retinanet	Smooth L1-Loss	71.52	45.62	41.98	74.18	48.39	43.39
	GWD	74.18	46.22	43.23	75.82	49.64	45.53
	KLD	74.41	47.05	43.12	76.41	50.41	45.30
	EGWD	74.62	46.91	43.45	76.86	50.92	45.69
	EDWD	**74.77**	**47.60**	**43.73**	**77.49**	**51.23**	**46.18**
FCOS	L1	70.71	43.01	41.15	74.01	47.64	43.27
	GWD	74.50	44.27	42.19	75.78	48.52	44.61
	KLD	74.71	**47.37**	42.25	76.12	**49.31**	44.53
	EGWD	74.31	45.89	43.72	75.65	47.99	44.70
	EDWD	**74.94**	46.29	**43.14**	**76.33**	49.28	**45.07**

Table 7: Comparisons on MSRA-TD500

Detector	Method	No Rotate Aug			Rotate Aug		
		Hmean50	Hmean75	Hmean50:95	Hmean50	Hmean75	Hmean50:95
Retinanet	Smooth L1-Loss	64.34	38.86	38.39	70.24	41.94	40.20
	GWD	66.47	43.61	39.96	72.66	43.72	42.31
	KLD	66.21	46.86	41.94	73.55	45.44	42.14
	EGWD	67.10	47.32	42.01	73.98	48.64	44.10
	EDWD	**67.32**	**48.65**	**42.55**	**74.13**	**49.14**	**45.08**

several categories that have large aspect ratio to report the performance. We use 5-AP_{50} to denote their average. The scene text datasets are also large aspect ratio dominated. It is shown that EWD loss gets comparable results with KLD loss on these targets. In most times, the EWD loss exhibits better performance compared with KLD loss.

Polynomial Regression To verify the performance on general polynomial regressions, we design a polynomial regression head based on FCOSTian et al. (2020). The regression branch regresses four offset vectors $(\delta x, \delta y)$ relative to the anchor point. The other part of the detector is kept identical to the original FCOS. For comparison, we choose smooth L1 loss, modulated lossQian et al. (2021a), gliding vertexXu et al. (2020). These methods are chosen for they can also be applied to the quadrilateral regression occasion. For smooth L1 loss, the regression target is sorted in the order of top point, right point, bottom point, and left point. Note that gliding vertex regresses a set of specially designed parameters, horizontal width/height, four gliding offset(referring to the original paper for details). We follow their setting and modify the regression head correspondingly. The comparison results are shown in Table-4. From the results, we find that for L1 loss, although the regression targets are sorted in order, the optimization is still unstable and tend to converge slowly. The modulated loss gets the second best results and the proposed EWD loss gets the best, even higher mAP(59.62vs55.32) than in the oriented case.

5 CONCLUSION

In this paper, we propose a novel distance measure based on the Wasserstein distance of bounding box's edges. By making some assumptions, we develop a novel loss function for oriented object detection, aka, Edge Wasserstein Distance loss (EWD). It is applicable to the general polynomial regression case, including the commonly seen oriented bonding box regression and quadrilateral regression. Interestingly, we find that for the oriented bounding box, the EWD loss turns out to be a generalized form of L2-distance loss. To verify the proposed method, extensive experiments are conducted on aerial images and scene text images. It demonstrates that the proposed method gets satisfactory results.

REFERENCES

Djalil Chafaï. Wasserstein distance between two gaussians. Website, 2010. https://djalil.chafai.net/blog/2010/04/30/wasserstein-distance-between-two-gaussians/.

Zhiming Chen, Kean Chen, Weiyao Lin, John See, Hui Yu, Yan Ke, and Cong Yang. Piou loss: Towards accurate oriented object detection in complex environments. In *European Conference on Computer Vision*, pp. 195–211. Springer, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Yingying Jiang, Xiangyu Zhu, Xiaobing Wang, Shuli Yang, Wei Li, Hua Wang, Pei Fu, and Zhenbo Luo. R2cnn: rotational region cnn for orientation robust scene text detection. *arXiv preprint arXiv:1706.09579*, 2017.

Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, et al. Icdar 2015 competition on robust reading. In *2015 13th International Conference on Document Analysis and Recognition*, pp. 1156–1160. IEEE, 2015.

Minghui Liao, Zhaoyi Wan, Cong Yao, Kai Chen, and Xiang Bai. Real-time scene text detection with differentiable binarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 11474–11481, 2020.

T. Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017a.

T. Y. Lin, P. Goyal, R. Girshick, K. He, and P Dollár. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, PP(99):2999–3007, 2017b.

Lei Liu, Zongxu Pan, and Bin Lei. Learning a rotation invariant detector with rotatable bounding box. *arXiv preprint arXiv:1711.09405*, 2017a.

Zikun Liu, Liu Yuan, Lubin Weng, and Yiping Yang. A high resolution optical satellite image dataset for ship recognition and some new baselines. In *Proceedings of the International Conference on Pattern Recognition Applications and Methods*, volume 2, pp. 324–331, 2017b.

Shangbang Long, Jiaqiang Ruan, Wenjie Zhang, Xin He, Wenhao Wu, and Cong Yao. Textsnake: A flexible representation for detecting text of arbitrary shapes. In *European Conference on Computer Vision*, pp. 20–36, 2018.

Wen Qian, Xue Yang, Silong Peng, Junchi Yan, and Yue Guo. Learning modulated loss for rotated object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 2458–2466, 2021a.

Wen Qian, Xue Yang, Silong Peng, Junchi Yan, and Xiujuan Zhang. Rsdet++: Point-based modulated loss for more accurate rotated object detection. *arXiv preprint arXiv:2109.11906*, 2021b.

Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 658–666, 2019.

Xuepeng Shi, Shiguang Shan, Meina Kan, Shuzhe Wu, and Xilin Chen. Real-time rotation-invariant face detection with progressive calibration networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2295–2303, 2018.

Z. Tian, C. Shen, H. Chen, and T. He. Fcos: Fully convolutional one-stage object detection. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2020.

Jinwang Wang, Jian Ding, Haowen Guo, Wensheng Cheng, Ting Pan, and Wen Yang. Mask obb: A semantic attention-based mask oriented bounding box representation for multi-category object detection in aerial images. *Remote Sensing*, 11(24):2930, 2019.

Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Dota: A large-scale dataset for object detection in aerial images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3974–3983, 2018.

Enze Xie, Wenhai Wang, Ding Mingyu, Zhang Ruimao, and Ping Luo. Polarmask++: Enhanced polar representation for single-shot instance segmentation and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

Yongchao Xu, Mingtao Fu, Qimeng Wang, Yukang Wang, Kai Chen, Gui-Song Xia, and Xiang Bai. Gliding vertex on the horizontal bounding box for multi-oriented object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

Xue Yang and Junchi Yan. Arbitrary-oriented object detection with circular smooth label. In *European Conference on Computer Vision*, pp. 677–694. Springer, 2020.

Xue Yang, Hao Sun, Xian Sun, Menglong Yan, Zhi Guo, and Kun Fu. Position detection and direction prediction for arbitrary-oriented ships via multitask rotation region convolutional neural network. *IEEE Access*, 6:50839–50849, 2018.

Xue Yang, Jirui Yang, Junchi Yan, Yue Zhang, Tengfei Zhang, Zhi Guo, Xian Sun, and Kun Fu. Scrdet: Towards more robust detection for small, cluttered and rotated objects. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 8232–8241, 2019.

Xue Yang, Liping Hou, Yue Zhou, Wentao Wang, and Junchi Yan. Dense label encoding for boundary discontinuity free rotation detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 15819–15829, 2021a.

Xue Yang, Junchi Yan, Ming Qi, Wentao Wang, Zhang Xiaopeng, and Tian Qi. Rethinking rotated object detection with gaussian wasserstein distance loss. In *International Conference on Machine Learning*, 2021b.

Xue Yang, Xiaojiang Yang, Jirui Yang, Qi Ming, Wentao Wang, Qi Tian, and Junchi Yan. Learning high-precision bounding box for rotated object detection via kullback-leibler divergence. In *Advances in Neural Information Processing Systems*, 2021c.

Xue Yang, Yue Zhou, and Junchi Yan. Alpharotate: A rotation detection benchmark using tensorflow. 2021d. URL https://github.com/yangxue0827/RotationDetection.

Cong Yao, Xiang Bai, Wenyu Liu, Yi Ma, and Zhuowen Tu. Detecting texts of arbitrary orientations in natural images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1083–1090. IEEE, 2012.

Jiahui Yu, Yuning Jiang, Zhangyang Wang, Zhimin Cao, and Thomas Huang. Unitbox: An advanced object detection network. In *Proceedings of the 24th ACM international conference on Multimedia*, pp. 516–520, 2016.

Zhaohui Zheng, Ping Wang, Wei Liu, Jinze Li, Rongguang Ye, and Dongwei Ren. Distance-iou loss: Faster and better learning for bounding box regression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 12993–13000, 2020.

Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. East: an efficient and accurate scene text detector. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5551–5560, 2017.

Yue Zhou, Xue Yang, Gefan Zhang, Jiabao Wang, Yanyi Liu, Liping Hou, Xue Jiang, Xingzhao Liu, Junchi Yan, Chengqi Lyu, Wenwei Zhang, and Kai Chen. Mmrotate: A rotated object detection benchmark using pytorch. *arXiv preprint arXiv:2204.13317*, 2022.

A APPENDIX

A.1 FULL DERIVATION OF EDGE GAUSSIAN WASSERSTEIN DISTANCE

The Wasserstein distance between two Gaussian distributions is written as:

$$\mathbf{W_{12}} = ||\mu_1 - \mu_2||_2^2 + \mathbf{Tr}\left(\Sigma_1 + \Sigma_2 - 2(\Sigma_1^{1/2}\Sigma_2\Sigma_1^{1/2})^{1/2}\right) \tag{17}$$

where μ_i is the edge center point and Σ_i is the covariance matrix and is defined as

$$\begin{aligned}
\mathbf{\Sigma_i} &= \mathbf{R_i S_i^2 R_i}^\top \\
&= \begin{pmatrix} \cos\theta_i & -\sin\theta_i \\ \sin\theta_i & \cos\theta_i \end{pmatrix} \begin{pmatrix} \frac{w_i^2}{4} & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \cos\theta_i & \sin\theta_i \\ -\sin\theta_i & \cos\theta_i \end{pmatrix} \\
&= \begin{pmatrix} \frac{w_i^2}{4}\cos^2\theta_i & \frac{w_i^2}{4}\cos\theta_i\sin\theta_i \\ \frac{w_i^2}{4}\cos\theta_i\sin\theta_i & \frac{w_i^2}{4}\sin^2\theta_i \end{pmatrix}
\end{aligned} \tag{18}$$

where w represents the edge's length and θ_i represents the angle of the edge. Then, for the latter form of Eq-17, it can be simplified by

$$\begin{aligned}
\mathbf{Tr}(...) &= \mathbf{Tr}(\mathbf{\Sigma_1}) + \mathbf{Tr}(\mathbf{\Sigma_2}) - 2 * \mathbf{Tr}((\Sigma_1^{1/2}\Sigma_2\Sigma_1^{1/2})^{1/2}) \\
&= \frac{w_1^2}{4} + \frac{w_2^2}{4} - 2 * \mathbf{Tr}((\Sigma_1^{1/2}\Sigma_2\Sigma_1^{1/2})^{1/2})
\end{aligned} \tag{19}$$

Then, for the term $((\Sigma_1^{1/2}\Sigma_2\Sigma_1^{1/2})^{1/2}$, note that $\Sigma_i^{1/2} = R_i S_i R_i^\top$, we have

$$\begin{aligned}
(\Sigma_1^{1/2}\Sigma_2\Sigma_1^{1/2})^{1/2} &= \left(R_1 S_1 R_1^\top R_2 S_2 S_2 R_2^\top R_1 S_1 R_1^\top\right)^{1/2} \\
&= \left(R_1 \left(S_1 R_1^\top R_2 S_2\right) \left(S_1 R_1^\top R_2 S_2\right)^\top R_1^\top\right)^{1/2} \\
&= \left(R_1 \begin{pmatrix} \frac{1}{4}w_1 w_2(\cos\theta_1\cos\theta_2 + \sin\theta_1\sin\theta_2) & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{4}w_1 w_2(\cos\theta_1\cos\theta_2 + \sin\theta_1\sin\theta_2) & 0 \\ 0 & 0 \end{pmatrix}^\top R1^\top\right)^{1/2} \\
&= R_1 \begin{pmatrix} \frac{1}{4}w_1 w_2 \cos\Delta\theta & 0 \\ 0 & 0 \end{pmatrix} R_1^\top
\end{aligned} \tag{20}$$

Finally, the trace term is developed as

$$\begin{aligned}
\mathbf{Tr}(...) &= \frac{w_1^2}{4} + \frac{w_2^2}{4} - \frac{1}{2}w_1 w_2 \cos\Delta\theta \\
&= \frac{1}{4}||\Delta\mathbf{w}||^2
\end{aligned} \tag{21}$$

Hence, we get the simplified version of EGWD for a single edge pair as

$$\mathbf{W_{12}} = ||\Delta\mu||^2 + \frac{1}{4}||\Delta\mathbf{w}||^2 \tag{22}$$

There are two constraints on the condition of OBox. First, the center points of each edge can be expressed by the 5-p vector (x, y, w, h, θ). Second, the parallel edges share identical covariance. We use o to denote the box's center point, μ to denote the edge's center point, $\mathbf{w}$ to denote the vector corresponding to width and $\mathbf{h}$ to denote the vector corresponding to height. By clockwise order, the center points of each edge can be written as

$$\begin{aligned}
\mu_1 = o - \frac{\mathbf{h}}{2}, \mu_2 = o + \frac{\mathbf{w}}{2} \\
\mu_3 = o + \frac{\mathbf{h}}{2}, \mu_4 = o - \frac{\mathbf{w}}{2}
\end{aligned} \tag{23}$$

Then, the EGWD can be developed as

$$\begin{aligned}
\mathbf{W}(p, q) &= ||\Delta(o - \frac{\mathbf{h}}{2})||^2 + ||\Delta(o + \frac{\mathbf{w}}{2})||^2 + ||\Delta(o + \frac{\mathbf{h}}{2})||^2 + ||\Delta(o - \frac{\mathbf{w}}{2})||^2 \\
&\quad + \frac{1}{2}||\Delta\mathbf{w}||^2 + \frac{1}{2}||\Delta\mathbf{h}||^2 \\
&= 4||\Delta o||^2 + ||\Delta\mathbf{w}||^2 + ||\Delta\mathbf{h}||^2
\end{aligned} \tag{24}$$

A.2 Full Derivation of Edge Dense Wasserstein Distance

We denote the edge center point as $\mathbf{c}$, the edge vector as $\mathbf{w}$. For a pair of edge, suppose two sampled points $\mathbf{X}_1$, $\mathbf{X}_2$ from two edges can be denoted as

$$\mathbf{X}_i = \mathbf{c} + x\frac{\mathbf{w_i}}{2} \tag{25}$$

where $x \in [-1, 1]$ Then the two edge's distance can be written as

$$
\begin{aligned}
\mathbf{W_{12}} &= \int_x p(x)||\Delta\mathbf{c} + x\frac{\Delta\mathbf{w}}{2}||^2 dx \\
&= ||\Delta\mathbf{c}||^2 + \int_x p(x)x^2\frac{||\Delta\mathbf{w}||^2}{4}dx + \int_x p(x)x\Delta\mathbf{c}\Delta\mathbf{w}dx
\end{aligned}
\tag{26}
$$

In this paper, we choose the probability density function $p(x)$ to be axial symmetric around the edge center. Thus, the last term of Eq-26 is zero. Note that $\int_x p(x)x^2 dx$ represents the variance of the distribution, which we can use σ^2 to denote it. Finally, the expression can be written as

$$\mathbf{W_{12}} = ||\Delta\mathbf{c}||^2 + \frac{\sigma^2}{4}||\Delta\mathbf{w}||^2 \tag{27}$$

Combining Eq-23 and Eq-27, we get the formulation for OBox

$$\mathbf{W(p,q)} = 4||\mathbf{\Delta o}||^2 + \left(\frac{1}{2} + \frac{\sigma_\mathbf{w}^2}{2}\right)||\mathbf{\Delta w}||^2 + \left(\frac{1}{2} + \frac{\sigma_\mathbf{h}^2}{2}\right)||\mathbf{\Delta h}||^2 \tag{28}$$