A. Proof of $d := \mathcal{W}(\mathcal{N}(\mathbf{m}_1, \Sigma_1); \mathcal{N}(\mathbf{m}_2, \Sigma_2))$

The entire proof process refers to this blog (Chafai, 2010).

The Wasserstein coupling distance $\mathcal{W}$ between two probability measures $\mu$ and $\nu$ on $\mathbb{R}^n$ expressed as follows:

$$
\mathcal{W}(\mu; \nu) := \inf \mathbb{E}(\|\mathbf{X} - \mathbf{Y}\|^2_{\mathcal{F}}^{1/2})
$$

where the infimum runs over all random vectors $(\mathbf{X}, \mathbf{Y})$ of $\mathbb{R}^n \times \mathbb{R}^n$ with $\mathbf{X} \sim \mu$ and $\mathbf{Y} \sim \nu$. It turns out that we have the following formula for $d := \mathcal{W}(\mathcal{N}(\mathbf{m}_1, \Sigma_1); \mathcal{N}(\mathbf{m}_2, \Sigma_2))$:

$$
d^2 = \|\mathbf{m}_1 - \mathbf{m}_2\|^2_2 + \text{Tr} \left( \Sigma_1 + \Sigma_2 - 2(\Sigma_1^{1/2} \Sigma_2^{1/2})^{1/2} \right)
$$

This formula interested several works (Givens et al., 1984; Olkin & Pukelsheim, 1982; Knott & Smith, 1984; Dowson & Landau, 1982). Note in particular we have:

$$
\text{Tr} \left( (\Sigma_1^{1/2} \Sigma_2^{1/2})^{1/2} \right) = \text{Tr} \left( (\Sigma_2^{1/2} \Sigma_1^{1/2})^{1/2} \right)
$$

In the commutative case $\Sigma_1 \Sigma_2 = \Sigma_2 \Sigma_1$, Eq. 2 becomes:

$$
d^2 = \|\mathbf{m}_1 - \mathbf{m}_2\|^2_2 + \|\Sigma_1^{1/2} - \Sigma_2^{1/2}\|^2_2
$$

$$
= (x_1 - x_2)^2 + (y_1 - y_2)^2 + \frac{(w_1 - w_2)^2 + (h_1 - h_2)^2}{4}
$$

$$
= ||\mathbf{f}(w, h)||_2
$$

where $||\cdot||_F$ is the Frobenius norm. Note that both boxes are horizontal at this time, and Eq. 4 is approximately equivalent to the $l_2$-norm loss (note the additional denominator of 2 for $w$ and $h$), which is consistent with the loss commonly used in horizontal detection. This also partly proves the correctness of using Wasserstein distance as the regression loss.

To prove Eq. 2, one can first reduce to the centered case $\mathbf{m}_1 = \mathbf{m}_2 = \mathbf{0}$. Next, if $(\mathbf{X}, \mathbf{Y})$ is a random vector (Gaussian or not) of $\mathbb{R}^n \times \mathbb{R}^n$ with covariance matrix

$$
\Gamma = \begin{pmatrix} \Sigma_1 & C \\ C^T & \Sigma_2 \end{pmatrix}
$$

then the quantity

$$
\mathbb{E}(\|\mathbf{X} - \mathbf{Y}\|^2_{\mathcal{F}}) = \text{Tr}(\Sigma_1 + \Sigma_2 - 2\mathbf{C})
$$

depends only on $\Gamma$. Also, when $\mu = \mathcal{N}(\mathbf{0}, \Sigma_1)$ and $\nu = \mathcal{N}(\mathbf{0}, \Sigma_2)$, one can restrict the infimum which defines $\mathcal{W}$ to run over Gaussian laws $\mathcal{N}(\mathbf{0}, \Gamma)$ on $\mathbb{R}^n \times \mathbb{R}^n$ with covariance matrix $\Gamma$ structured as above. The sole constrain on $\mathbf{C}$ is the Schur complement constraint:

$$
\Sigma_1 - \mathbf{C}\Sigma_2^{-1}\mathbf{C}^T \succeq 0
$$

The minimization of the function

$$
\mathbf{C} \mapsto -2\text{Tr}(\mathbf{C})
$$

under the constraint above leads to Eq. 2. A detailed proof is given by (Givens et al., 1984). Alternatively, one may find an optimal transportation map as (Knott & Smith, 1984). It
To check that the map is optimal, one may use, 

$$f(\cdot) = \text{sqrt}$$ 

$$f(\cdot) = \log$$ 

<table>
<thead>
<tr>
<th>$1 - \frac{1}{\tau + f(d^2)}$</th>
<th>$\tau = 1$</th>
<th>$\tau = 2$</th>
<th>$\tau = 3$</th>
<th>$\tau = 5$</th>
<th>$f(d^2)$</th>
<th>$d^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f(\cdot) = \text{sqrt}$</td>
<td>68.56</td>
<td>68.93</td>
<td>68.37</td>
<td>67.17</td>
<td>54.27</td>
<td>49.11</td>
</tr>
<tr>
<td>$f(\cdot) = \log$</td>
<td>67.87</td>
<td>68.09</td>
<td>67.48</td>
<td>66.49</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

turns out that $N^T(m_2, \Sigma_2)$ is the image law of $N^T(m_1, \Sigma_1)$ with the linear map 

$$x \mapsto m_2 + A(xm_1)$$

where 

$$A = \Sigma_1^{-1/2} (\Sigma_1^{1/2} \Sigma_2 \Sigma_1^{1/2})^{1/2} \Sigma_1^{-1/2} = A^T$$

(10)

To check that this maps $N^T(m_1, \Sigma_1)$ to $N^T(m_2, \Sigma_2)$, say in the case $m_1 = m_2 = 0$ for simplicity, one may define the random column vectors $X \sim N^T(m_1, \Sigma_1)$ and $Y = AX$ and write 

$$E(YY^T) = E(XX^T)A^T$$

$$= \Sigma_1^{1/2} (\Sigma_1^{1/2} \Sigma_2 \Sigma_1^{1/2})^{1/2} (\Sigma_1^{1/2} \Sigma_2 \Sigma_1^{1/2})^{1/2} \Sigma_1^{1/2} = \Sigma_2$$

(11)

To check that the map is optimal, one may use, 

$$E(\|X - Y\|^2) = E(\|X\|^2) + E(\|Y\|^2) - 2E(\langle X, Y \rangle)$$

$$= \text{Tr}(\Sigma_1) + \text{Tr}(\Sigma_2) - 2E(\langle X, AX \rangle)$$

$$= \text{Tr}(\Sigma_1) + \text{Tr}(\Sigma_2) - 2\text{Tr}(\Sigma_1 A)$$

(12)

and observe that by the cyclic property of the trace, 

$$\text{Tr}(\Sigma_1 A) = \text{Tr}((\Sigma_1^{1/2} \Sigma_2 \Sigma_1^{1/2})^{1/2})$$

(13)

The generalizations to elliptic families of distributions and to infinite dimensional Hilbert spaces is probably easy. Some more “geometric” properties of Gaussians with respect to such distances where studied more recently by (Takatsu & Yokota, 2012) and (Takatsu & Yokota, 2012).

**B. Supplementary Experiment**

**B.1. Improved GWD-based Regression Loss**

In Tab. 1, we compare three different forms of GWD-based regression loss, including $d^2$, $1 - \frac{1}{\tau + f(d^2)}$ and $f(d^2)$. The performance of directly using GWD ($d^2$) as the regression loss is extremely poor, only 49.11%, due to its rapid growth trend (as shown on the left of Fig. 1). In other words, the regression loss $d^2$ is too sensitive to large errors. In contrast, $1 - \frac{1}{\tau + f(d^2)}$ achieves a significant improvement by fitting IoU loss. This loss form introduces two new hyperparameters, the non-linear function $f(\cdot)$ to transform the Wasserstein distance, and the constant $\tau$ to modulate the entire loss. From Tab. 1, the overall performance of using sqrt outperforms that using log, about 0.98±0.3% higher. For $f(\cdot) = \text{sqrt}$ with $\tau = 2$, the model achieves the best performance, about 68.93%. In order to further reduce the number of hyperparameters of the loss function, we directly use the GWD after nonlinear transformation ($f(d^2)$) as the regression loss. As shown in the red box in Fig. 1, $f(d^2)$ still has a nearly linear trend after transformation using the nonlinear function sqrt and only achieves 54.27%. In comparison, the log function can better make the $f(d^2)$ change value close to IoU loss (see green box in Fig. 1) and achieve the highest performance, about 69.82%. In general, we do not need to strictly fit the IoU loss, and the regression loss should not be sensitive to large errors.

**B.2. Training Strategies and Tricks**

In order to further improve the performance of the model on DOTA, we verified many commonly used training strategies and tricks, including backbone, training schedule, data augmentation (DA), multi-scale training and testing (MS), stochastic weights averaging (SWA) (Izmailov et al., 2018; Zhang et al., 2020), multi-scale image cropping (MSC), model ensemble (ME), as shown in Tab. 2.

**Backbone:** Under the conditions of different detectors (RetinaNet and R^3Det), different training schedules (experimental groups $\{#16,#21\}$, $\{#24,#29\}$), and different tricks (experimental groups $\{#26,#31\}$, $\{#28,#33\}$), large backbone can bring stable performance improvement.

**Multi-scale training and testing:** Multi-scale training and testing is an effective means to improve the performance of aerial images with various object scales. In this paper, training and testing scale set to [450, 500, 640, 700, 800, 900, 1,000, 1,100, 1,200]. Experimental groups $\{#3,#4\}$, $\{#5,#6\}$ and $\{#11,#12\}$ show the its effectiveness, increased by 0.9%, 1.09%, and 0.58%, respectively.

**Training schedule:** When data augmentation and multi-scale training are added, it is necessary to appropriately lengthen the training time. From the experimental groups $\{#3,#5\}$ and $\{#16,#29\}$, we can find that the performance respectively increases by 0.77% and 1.22% when the training schedule is increased from 40 or 30 epochs to 60 epochs.

**Stochastic weights averaging (SWA):** SWA technique has been proven to be an effective tool for improving object detection. In the light of (Zhang et al., 2020), we train our detector for an extra 12 epochs using cyclical learning rates and then average these 12 checkpoints as the final detection model. It can be seen from experimental groups $\{#1,#2\}$, $\{#20,#21\}$ and $\{#25,#26\}$ in Tab. 2 that we get 0.99%, 1.20% and 1.13% improvement on the challenging DOTA benchmark.
Table 2. Ablation experiment of training strategies and tricks. R-101 denotes ResNet-101 (likewise for R-18, R-50, R-152). MS, MSC, SWA, and ME represent data augmentation, multi-scale training and testing, stochastic weights averaging, multi-scale image cropping, and model ensemble, respectively. The short names for categories are defined as (abbreviation-full name): PL-Plane, BD-Baseball diamond, BR-Bridge, GTF-Ground field track, SV-Small vehicle, LV-Large vehicle, SH-SHIP, TC-Tennis court, BC-Basketball court, ST-Storage tank, SBF-Soccer-ball field, RA-Roundabout, HA-Harbor, SP-Swimming pool, and HC-Helicopter.

<table>
<thead>
<tr>
<th>ID</th>
<th>METHOD</th>
<th>BACKGROUND</th>
<th>SCIRD</th>
<th>DA</th>
<th>MS</th>
<th>MSC</th>
<th>SWA</th>
<th>ME</th>
<th>PL</th>
<th>BD</th>
<th>GTF</th>
<th>SV</th>
<th>LV</th>
<th>SH</th>
<th>TC</th>
<th>SBF</th>
<th>RA</th>
<th>HA</th>
<th>SP</th>
<th>HC</th>
<th>MAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>R-50</td>
<td>20</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>11</td>
<td>R-101</td>
<td>30</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>21</td>
<td>R-152</td>
<td>60</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>31</td>
<td>RETINA-GWD</td>
<td>80</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>41</td>
<td>R-35</td>
<td>33</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>51</td>
<td>R-29</td>
<td>32</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>61</td>
<td>R-24</td>
<td>31</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>71</td>
<td>R-19</td>
<td>30</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>81</td>
<td>R-16</td>
<td>27</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>91</td>
<td>R-13</td>
<td>24</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>101</td>
<td>R-10</td>
<td>21</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>111</td>
<td>R-7</td>
<td>22</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

**Multi-scale image cropping**: Large-scene object detection often requires image sliding window cropping before training. During testing, sliding window cropping is required before the results are merged. Two adjacent sub-images often have an overlapping area to ensure that the truncated object can appear in a certain sub-image completely. The cropping size needs to be moderate, too large will cause large objects to be truncated with high probability. Multi-scale cropping is an effective detection technique that is beneficial to objects of various scales. In this paper, our multi-scale crop size and corresponding overlap size are [600, 800, 1,024, 1,300, 1,600] and [150, 200, 300, 400, 500], respectively. According to experimental groups #6, #7, the large object categories (e.g. GTF and SBF) that are often truncated have been significantly improved. Take group #6, #7 as an example, GTF and SBF increased by 6.43% and 6.14%, respectively.

**Results on DOTA**: Due to the complexity of the aerial image and the large number of small, cluttered and rotated objects, DOTA is a very challenging dataset. We compare the proposed approach with other state-of-the-art methods on DOTA, as shown in Tab. 3. As far as I know, this is the most comprehensive statistical comparison of methods on the DOTA dataset. Since different methods use different image resolution, network structure, training strategies and various tricks, we cannot make absolutely fair comparisons. In terms of overall performance, our method has achieved the best performance so far, at around 80.23%.

**Results on HRSC2016**: The HRSC2016 contains lots of large aspect ratio ship instances with arbitrary orientation, which poses a huge challenge to the positioning accuracy of the detector. Experimental results at Tab. 4 shows that our model achieves state-of-the-art performances, about 89.85% and 97.37% in term of 2007 and 2012 evaluation metric.

**References**


Table 4. Detection accuracy on HRSC2016.

<table>
<thead>
<tr>
<th>Method</th>
<th>Backbone</th>
<th>MAP06 (07)</th>
<th>MAP06 (12)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RetIN2c (Li et al., 2018)</td>
<td>VGG16</td>
<td>75.7</td>
<td>–</td>
</tr>
<tr>
<td>Axis Learning (Xiao et al., 2019)</td>
<td>R-101</td>
<td>78.15</td>
<td>–</td>
</tr>
<tr>
<td>TOSO (Feng et al., 2020)</td>
<td>R-101</td>
<td>79.29</td>
<td>–</td>
</tr>
<tr>
<td>RetIN2c (Li et al., 2018)</td>
<td>VGG16</td>
<td>79.6</td>
<td>–</td>
</tr>
<tr>
<td>RRD (Liao et al., 2018)</td>
<td>VGG16</td>
<td>84.3</td>
<td>–</td>
</tr>
<tr>
<td>RoI-Trans. (Ding et al., 2019)</td>
<td>R-101</td>
<td>86.20</td>
<td>–</td>
</tr>
<tr>
<td>RDC-Net (Qian et al., 2020)</td>
<td>R-101</td>
<td>84.6</td>
<td>–</td>
</tr>
<tr>
<td>CenterMap (Wang et al., 2020b)</td>
<td>R-50</td>
<td>86.20</td>
<td>–</td>
</tr>
<tr>
<td>Gliding Vertex (Xu et al., 2020b)</td>
<td>R-101</td>
<td>88.44</td>
<td>–</td>
</tr>
<tr>
<td>OPLD (Song et al., 2020)</td>
<td>R-50</td>
<td>88.44</td>
<td>–</td>
</tr>
<tr>
<td>S3Net (??)</td>
<td>R-101</td>
<td>90.17</td>
<td>95.01</td>
</tr>
<tr>
<td>RDet (Yang et al., 2020)</td>
<td>R-101</td>
<td>89.26</td>
<td>96.01</td>
</tr>
<tr>
<td>RDet-DC (??)</td>
<td>R-101</td>
<td>89.46</td>
<td>96.41</td>
</tr>
<tr>
<td>CNB (??)</td>
<td>R-101</td>
<td>86.82</td>
<td>96.41</td>
</tr>
</tbody>
</table>

Supplementary Materials

Table 3. AP on different objects and mAP on DOTA. R-101 denotes ResNet-101 (likewise for R-50, R-152), RX-101 and H-104 stands for ResNeXt101 (Xie et al., 2017) and Hourglass-104 (Newell et al., 2016). Other backbone include DPN-92 (Chen et al., 2017), DLA-34 (Yu et al., 2018), DCN (Dai et al., 2017), HRNet-W48 (Wang et al., 2020a), U-Net (Ronneberger et al., 2015). MS indicates that multi-scale training or testing is used.


Supplementary Materials


