A Two-Step Computation of the Exact GAN Wasserstein Distance Supplementary Material

Huidong Liu, Xianfeng David Gu, Dimitris Samaras

1. Proofs of Lemma 3.1 and 3.2

Lemma 3.1. If the cost function $c(\cdot,\cdot)$ satisfies the triangle inequality, i.e., $c(x,y)+c(y,z) \geq c(x,z), \forall x,y,z$, then $\forall x_j \in \hat{X},\ \forall y_i \in \hat{Y}$, if $x_j = y_i$, and ψ^* is the optimizer to Problem 3, then $(\psi^c)^*(y_i) = \psi^*(x_j)$, where $(\psi^c)^*(y_i) = \inf_{x\in\hat{X}} \big(\psi^*(x) + c(x,y_i)\big)$.

Proof. We prove this by contradiction. Without loss of generality, suppose x_s overlaps with y_t, i.e., $x_s = y_t$, and $(\psi^c)^*(y_1) \neq \psi^*(x_s)$. According to the definition of the c-transform in Eq. (5), $(\psi^c)^*(y_t) = \inf\{\psi^*(x_s), \inf_{x\in\hat{X}\setminus x_s} \psi^*(x) + c(x,y_t)\}$. Since $(\psi^c)^*(y_t) \neq \psi^*(x_s)$, for any y_i, we have

$$\begin{aligned} & \psi^*(x_s) + c(x_s,y_i) \\ > & \inf_{x\in\hat{X}\setminus x_s} (\psi^*(x) + c(x,y_t)) + c(x_s,y_i) \\ = & \inf_{x\in\hat{X}\setminus x_1} (\psi^*(x) + c(x,y_t) + c(x_s,y_i)) \\ \geq & \inf_{x\in\hat{X}\setminus x_s} (\psi^*(x) + c(x,y_i)) \end{aligned}$$

In the last step, we use that fact that $x_s = y_t$ and the triangle inequality of $c(\cdot,\cdot)$

$$\begin{aligned} & \hat{C}^*(\mu,\nu) \\ = & \frac{1}{m}\sum_{i\in\mathcal{I}} (\psi^c)^*(y_i) - \frac{1}{n}\sum_{j\in\mathcal{J}} \psi^*(x_j) \\ = & \frac{1}{m}\sum_{i\in\mathcal{I}} \inf_{x\in\hat{X}\setminus x_s} (\psi^*(x) + c(x,y_i)) - \frac{1}{n}\sum_{j\in\mathcal{J}} \psi^*(x_j) \end{aligned}$$

We can always find another function ψ', such that $\psi'(x) = \psi^*(x), \forall x \in \hat{X}\setminus x_s$, and $\psi^*(x_s) > \psi'(x_s) > \inf_{x\in\hat{X}\setminus x_s}(\psi^*(x) + c(x,y_t))$. In this case $(\psi^c)'(y_i) = (\psi^c)^*(y_i), \forall i \in \mathcal{I}$, but $\psi^*(x_s) > \psi'(x_s)$. So, $\hat{d}(\psi') > \hat{d}(\psi^*)$, a contradiction. □

Lemma 3.2 Suppose f^* is an optimizer to Problem 4, then i) $f^*(y) = \inf_{x\in\hat{X}}\{f^*(x) + c(x,y)\},\ \forall y \in \hat{Y}$, and ii) $f^*(x) = \sup_{y\in\hat{Y}}\{f^*(y) - c(x,y)\},\ \forall x \in \hat{X}$

Proof. i) Since f^* is the optimal solution to Problem 4, we have $f^*(y) \leq \inf_{x\in\hat{X}}\{f^*(x)+c(x,y)\},\ \forall y \in \hat{Y}$. We prove i) by contradiction. Suppose there exists a y_i, without loss of generality, say y_t, such that $f^*(y_t) < \inf_{x\in\hat{X}}\{f^*(x)+c(x,y_t)\}$ (Note that in this case y_t can not equal any x_j in $\hat{X}$. If, without loss of generality, y_t equals, say x_s, then we have $f^*(y_t) = \{f^*(x_s)+c(x_s,y_t)\} \geq \inf_{x\in\hat{X}}\{f^*(x)+c(x,y_t)\}$). There exists another function f' such that $f'(x_j) = f^*(x_j), \forall x_j \in \hat{X}$, $f'(y_i) = f^*(y_i), \forall y_i \in \hat{Y}\setminus y_t$ and $f'(y_t) = \inf_{x\in\hat{X}}\{f^*(x) + c(x,y_t)\}$. It is easy to verify that f' satisfies the constraints in Problem 4 and $\hat{h}(f') > \hat{h}(f^*)$, which leads to a contradiction. Therefore, $f^*(y) = \inf_{x\in\hat{X}}\{f^*(x) + c(x,y)\},\ \forall y \in \hat{Y}$.

ii) The proof is similar to i).

□

2. Proof of Theorem 3.5.

Theorem 3.5. Suppose f is the optimal solution to formula (8) with 0 optimization error, i.e., 0-suboptimal. Let θ_{ij} be a Bernoulli random variable such that if $f(y_i) - f(x_j) > c(y_i, x_j)$ then $\theta_{ij} = 1$, otherwise $\theta_{ij} = 0$. Let $e = \mathbb{E}[\theta_{ij}]$ be the expectation of the probability that constraints in Problem 4 violate the inequality constraints. The error bound of the Wasserstein distance from the discrete to the continuous case is:

$$P(|\hat{h}(f) - h(f)| > \epsilon) \leq 2\exp(-m\epsilon^2/2) + 2\exp(-n\epsilon^2/2)$$

The error bound of the constraint violation in Problem 4 is:

$$P\left(|e| > \epsilon\right) \leq 2\exp(-2mn\epsilon^2)$$

Proof. According to Hoeffding's inequality:

$$P\left(\left|\frac{1}{m}\frac{1}{n}\sum_{ij}\theta_{ij} - e\right| > \epsilon\right) \leq 2\exp(-2mn\epsilon^2)$$

Suppose we have a deep neural network with sufficient capacity that can solve the deep regression problem. Then, $f(y_i) - f(x_j) \leq c(y_i, x_j)$, $\forall x \in \hat{X}$ and $\forall y \in \hat{Y}$ and $\theta_{ij} = 0$. Therefore,

$$P\left(|e| > \epsilon\right) \leq 2\exp(-2mn\epsilon^2)$$

Next we prove that $|\hat{h}(f) - h(f)|$ is bounded:

$$\begin{aligned}
& |\hat{h}(f) - h(f)| \\
= & \left|\left(\frac{1}{m}\sum_{i=1}^{m} f(y_i) - \frac{1}{n}\sum_{j=1}^{n} f(x_j)\right) - (\mathbb{E}[f(y)] - \mathbb{E}[f(x)])\right| \\
= & \left|\left(\frac{1}{m}\sum_{i=1}^{m} f(y_i) - \mathbb{E}[f(y)]\right) - \left(\frac{1}{n}\sum_{j=1}^{n} f(x_j) - \mathbb{E}[f(x)]\right)\right| \\
\leq & \left|\left(\frac{1}{m}\sum_{i=1}^{m} f(y_i) - \mathbb{E}[f(y)]\right)\right| + \left|\left(\frac{1}{n}\sum_{j=1}^{n} f(x_j) - \mathbb{E}[f(x)]\right)\right|
\end{aligned} \tag{1}$$

According to Hoeffding's inequality:

$$P\left(\left|\left(\frac{1}{m}\sum_{i=1}^{m} f(y_i) - \mathbb{E}[f(y)]\right)\right| > \epsilon\right) \leq 2\exp(-2m\epsilon^2)$$

$$P\left(\left|\left(\frac{1}{n}\sum_{j=1}^{n} f(x_j) - \mathbb{E}[f(x)]\right)\right| > \epsilon\right) \leq 2\exp(-2n\epsilon^2)$$

$$\begin{aligned}
& P\left(\left|\left(\frac{1}{m}\sum_{i=1}^{m} f(y_i) - \mathbb{E}[f(y)]\right)\right| > \epsilon \bigcup \left|\left(\frac{1}{n}\sum_{j=1}^{n} f(x_j) - \mathbb{E}[f(x)]\right)\right| > \epsilon\right) \\
\leq & P\left(\left|\left(\frac{1}{m}\sum_{i=1}^{m} f(y_i) - \mathbb{E}[f(y)]\right)\right| > \epsilon\right) + P\left(\left|\left(\frac{1}{n}\sum_{j=1}^{n} f(x_j) - \mathbb{E}[f(x)]\right)\right| > \epsilon\right) \\
\leq & 2\exp(-2m\epsilon^2) + 2\exp(-2n\epsilon^2)
\end{aligned}$$

Therefore,

$$
\begin{aligned}
& P\left(\left|\left(\frac{1}{m}\sum_{i=1}^{m} f(y_i) - \mathbb{E}[f(y)]\right)\right| \leq \epsilon \bigcap \left|\left(\frac{1}{n}\sum_{j=1}^{n} f(x_j) - \mathbb{E}[f(x)]\right)\right| \leq \epsilon\right) \\
> \quad & 1 - 2\exp(-2m\epsilon^2) - 2\exp(-2n\epsilon^2)
\end{aligned}
$$

Let $\epsilon \leftarrow \epsilon/2$. We have

$$
\begin{aligned}
& P\left(\left|\left(\frac{1}{m}\sum_{i=1}^{m} f(y_i) - \mathbb{E}[f(y)]\right)\right| + \left|\left(\frac{1}{n}\sum_{j=1}^{n} f(x_j) - \mathbb{E}[f(x)]\right)\right| \leq \epsilon\right) \\
> \quad & 1 - 2\exp(-m\epsilon^2/2) - 2\exp(-n\epsilon^2/2)
\end{aligned}
$$

Therefore,

$$
P(|\hat{h}(f) - h(f)| > \epsilon) \leq 2\exp(-m\epsilon^2/2) + 2\exp(-n\epsilon^2/2)
$$

□

3. Workflow of the Proposed Method

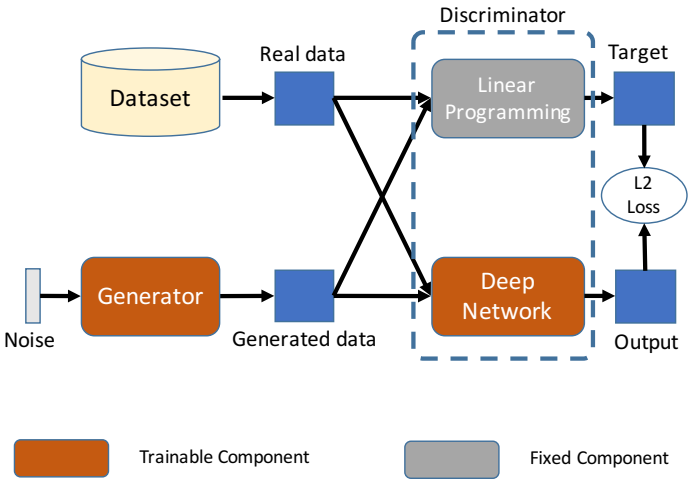

Figure 1. The workflow of the proposed WGAN-TS.

Figure 1 shows the workflow of WGAN-TS. The critic contains the Linear Programming part and the Deep Network part. The output of the linear programming part is compared with the output of the deep network part using an L2 loss. In the discriminator, only the deep network part performs back propagation.

The linear programming part computes the exact Wasserstein distance between the real data distribution and the generated data distribution. The Deep Network takes the same inputs and regresses the values produced by the linear programming part, thus providing a differentiable approximation of the Wasserstein distance.

4. The Eight Gaussian Toy Dataset

Dataset: We generate 8 Gaussians as the real data distribution. The centers of the 8 Gaussians are $(10, 0)$, $(-10, 0)$, $(0, 10)$, $(0, -10)$, $(10/\sqrt{2}, 10/\sqrt{2})$, $(10/\sqrt{2}, -10/\sqrt{2})$, $(-10/\sqrt{2}, 10/\sqrt{2})$ and $(-10/\sqrt{2}, -10/\sqrt{2})$. For each Gaussian, we generate 32 data points. Thus in total, we have 256 real data points. The synthetic data distribution is generated from

a Gaussian centered at $(0, 0)$ with $\sigma^2 = 1$. We sample 256 data points from the synthetic data distribution. Therefore, in total, we have 512 data points.

Parameter settings: We let the critics of all the methods iterate 2500 times. We use Adam as the optimizer and the learning rate of all methods is set to 0.01, $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The hyper-parameter c in WGAN is set to 0.01 and λ in WGAN-GP is set to 0.1 as suggested.

5. Large Version of Figures Shown in Paper

Figure 2. Large version of Figure 3(a) (WGAN)

Figure 3. Large version of Figure 3(b) (WGAN-GP)

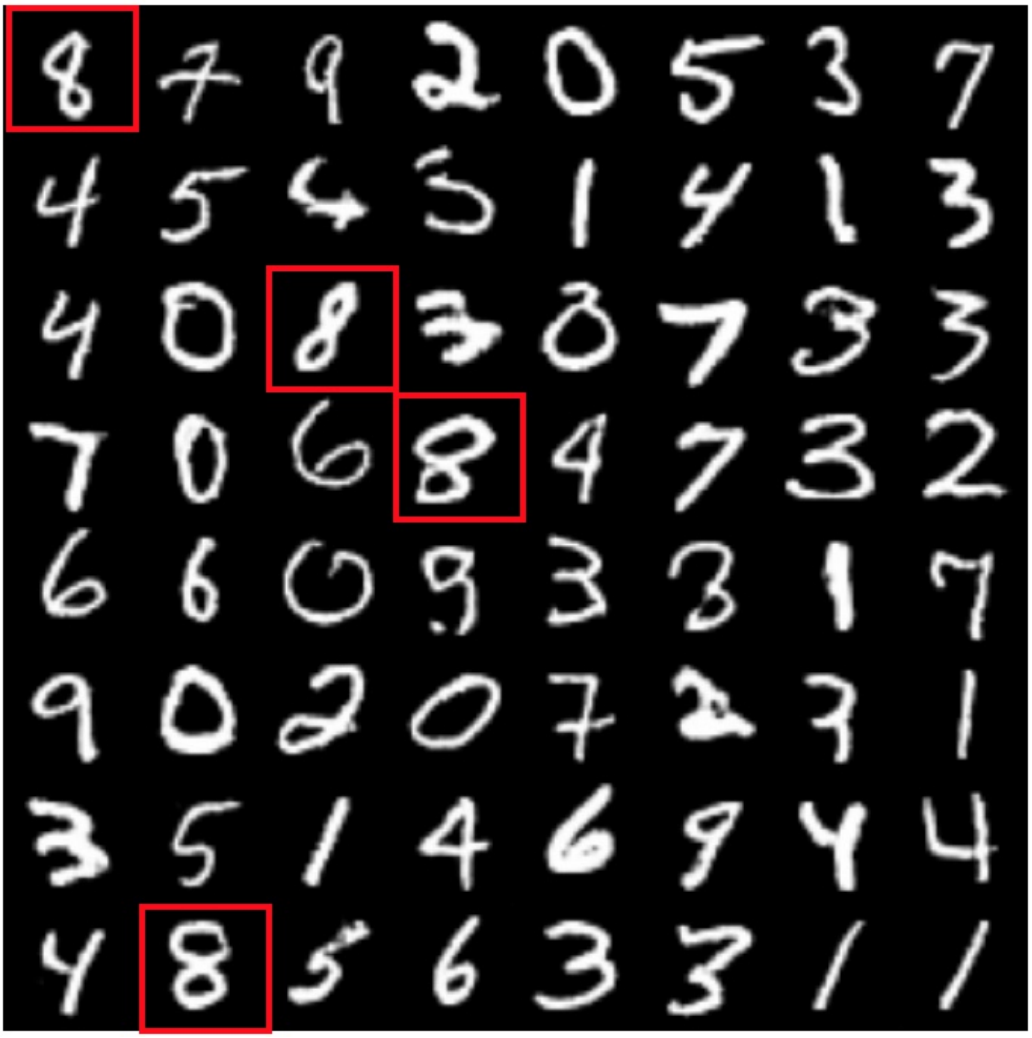

Figure 4. Large version of Figure 3(c) (SN-WD)

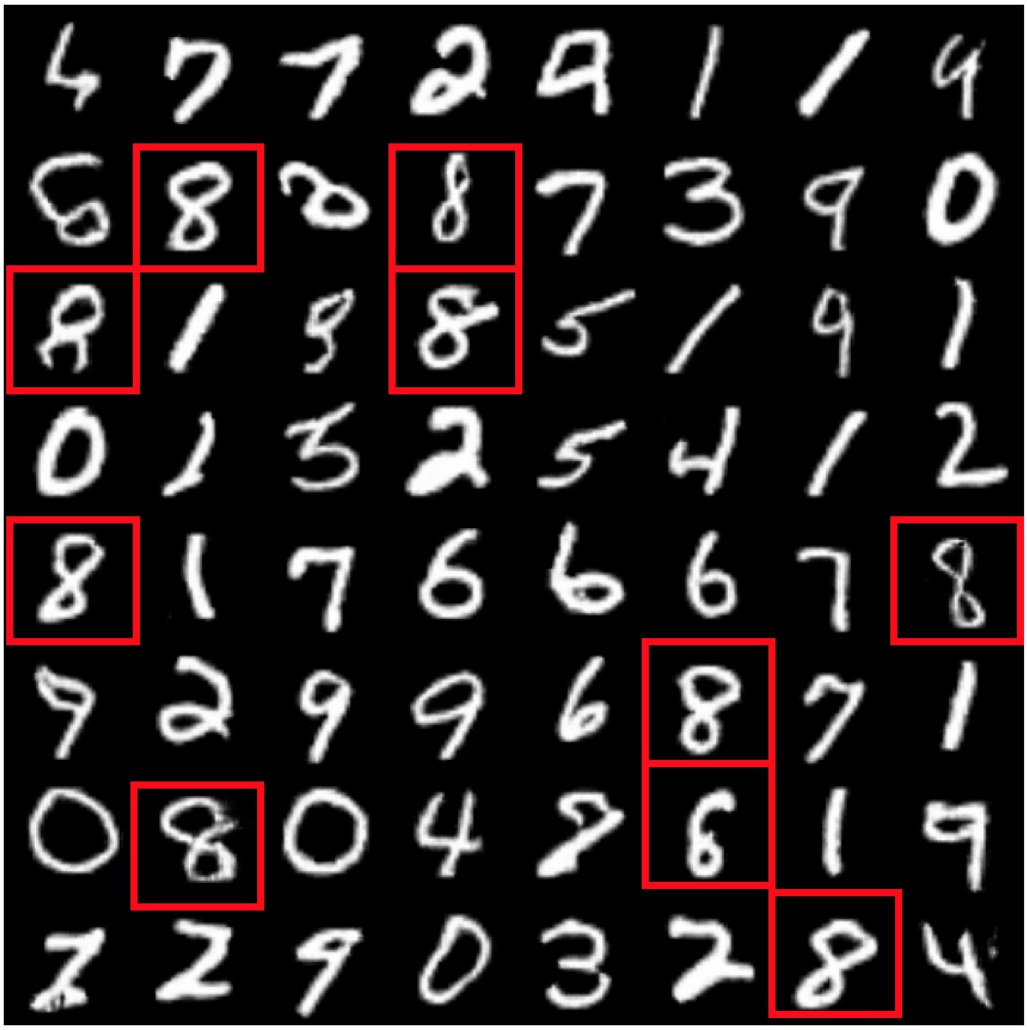

Figure 5. Large version of Figure 3(d) (WGAN-TS)

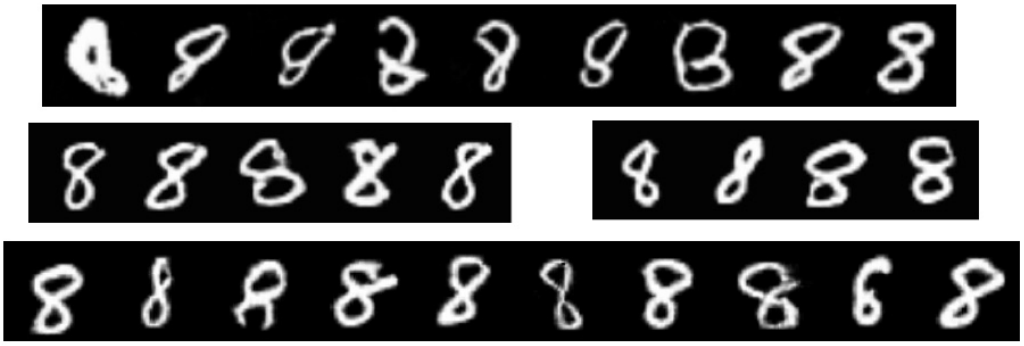

Figure 6. 1st row, images of digit 8 selected in Figure 2. 2nd row (left), images of 8 selected in Figure 3. 2nd row (right), images of 8 selected in Figure 4. 3rd row, images of digit 8 selected in Figure 5. WGAN-GP, SN-WD and WGAN-TS generate more realistic images of digit 8.

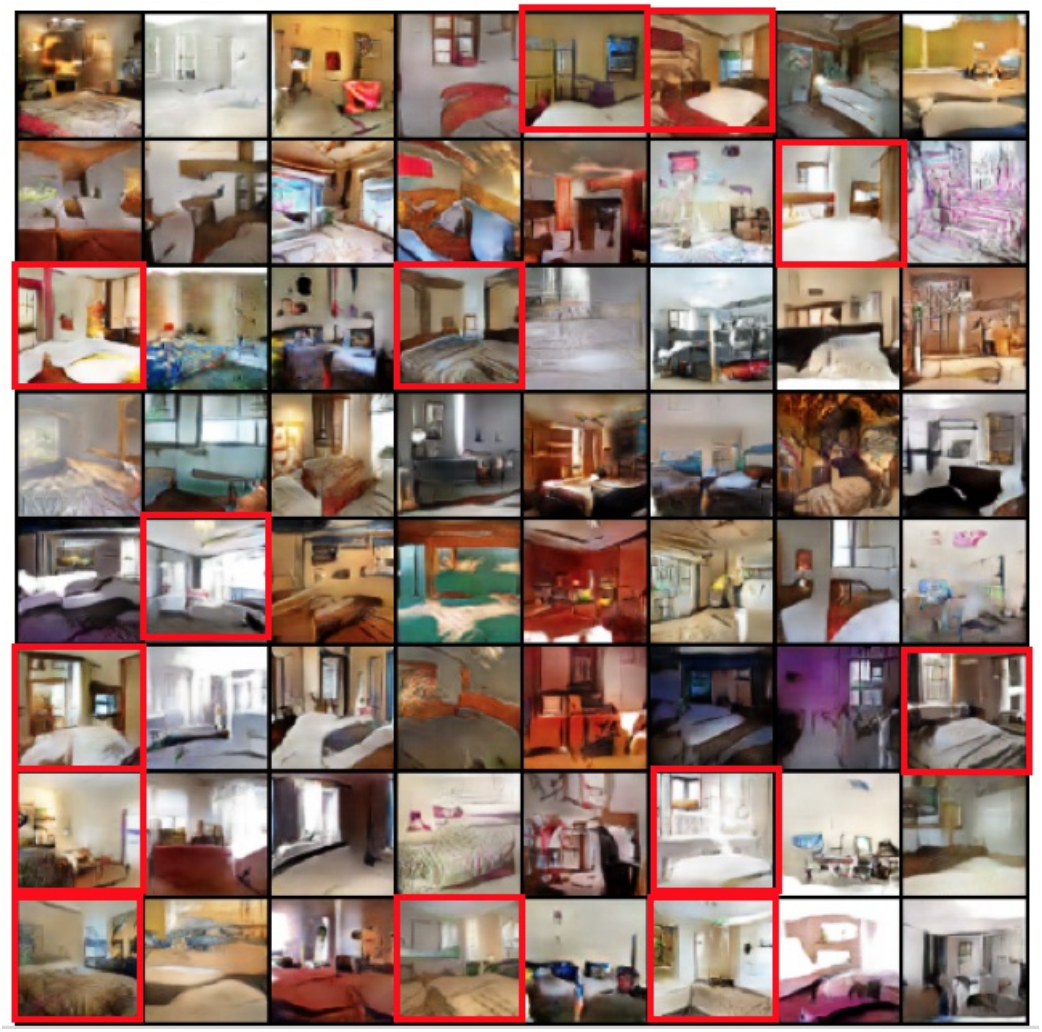

Figure 7. Large version of Figure 4 (a) (WGAN)

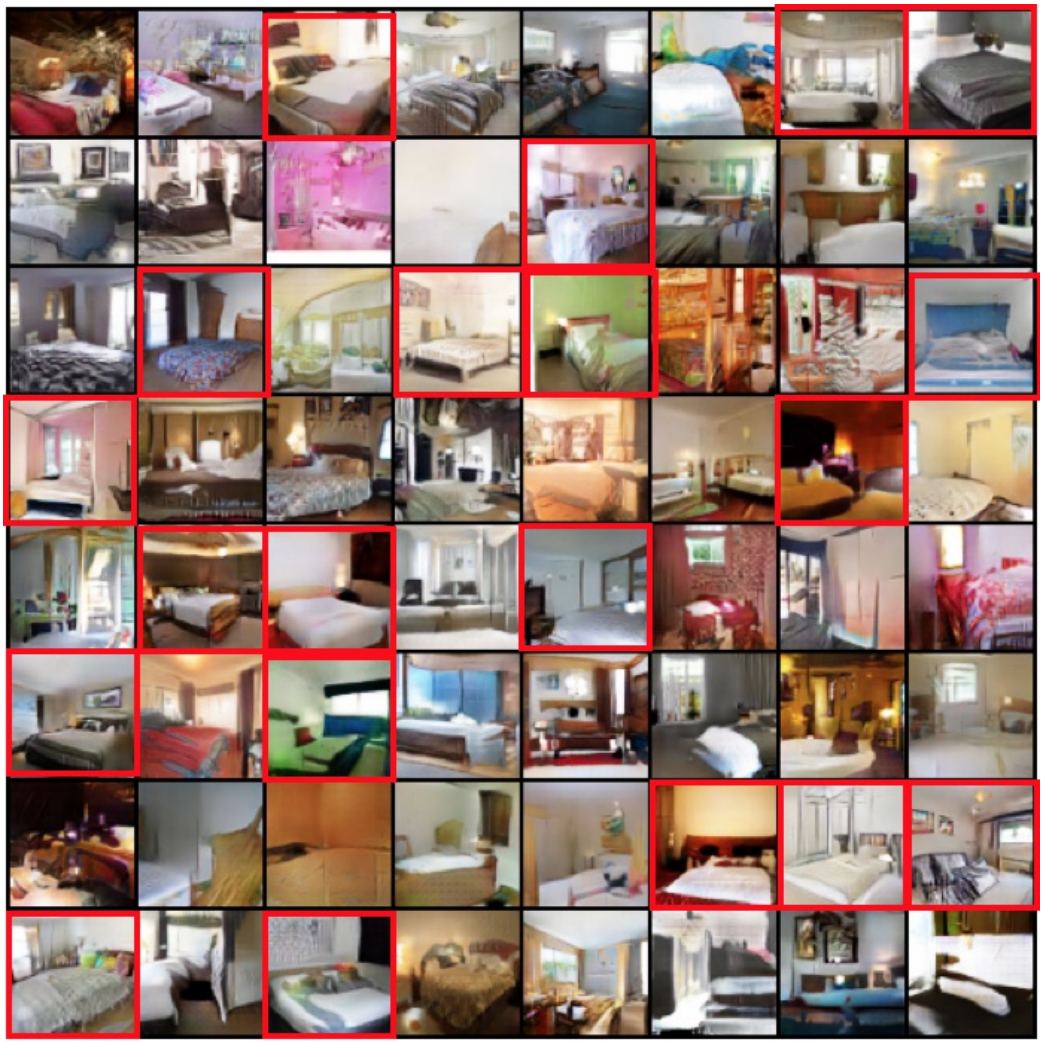

Figure 8. Large version of Figure 4 (b) (WGAN-GP)

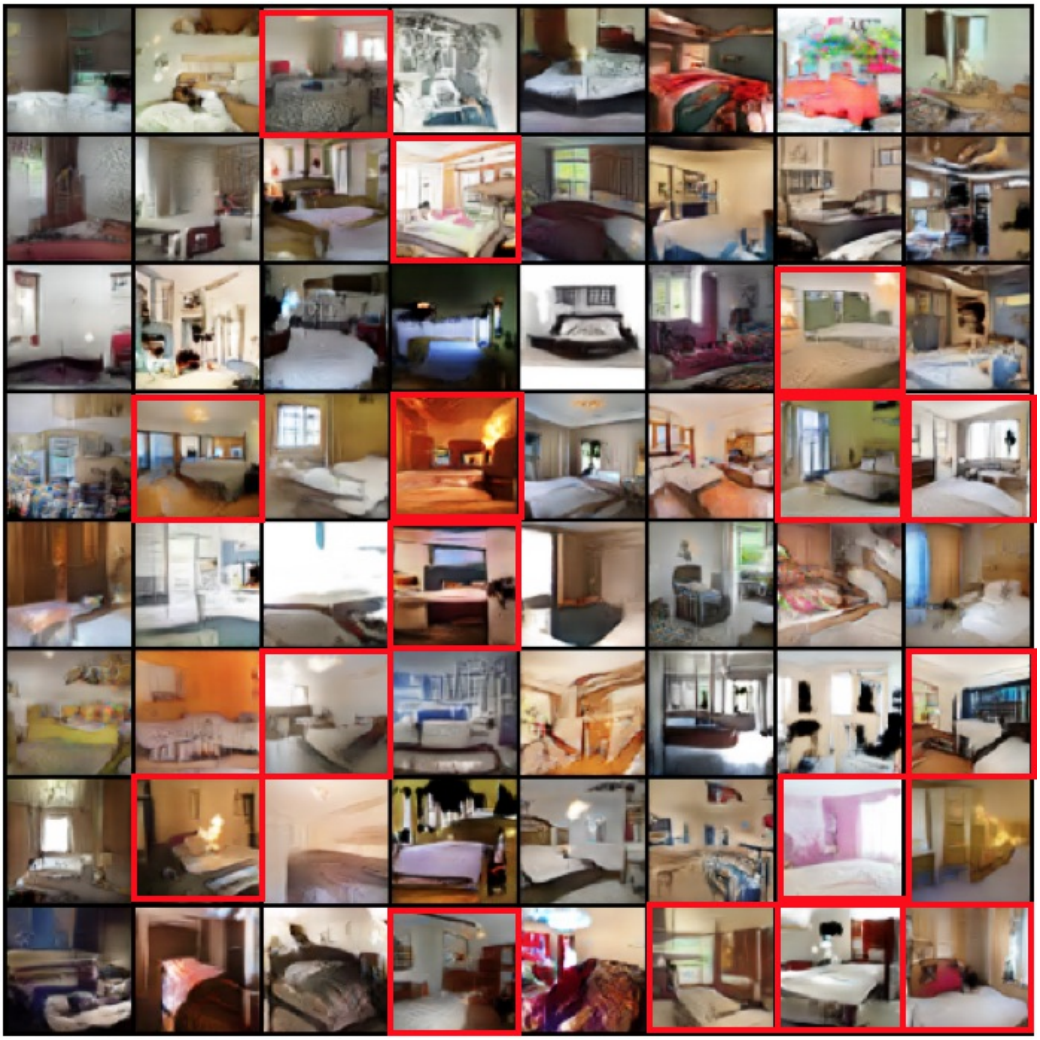

Figure 9. Large version of Figure 4 (c) (SN-WD)

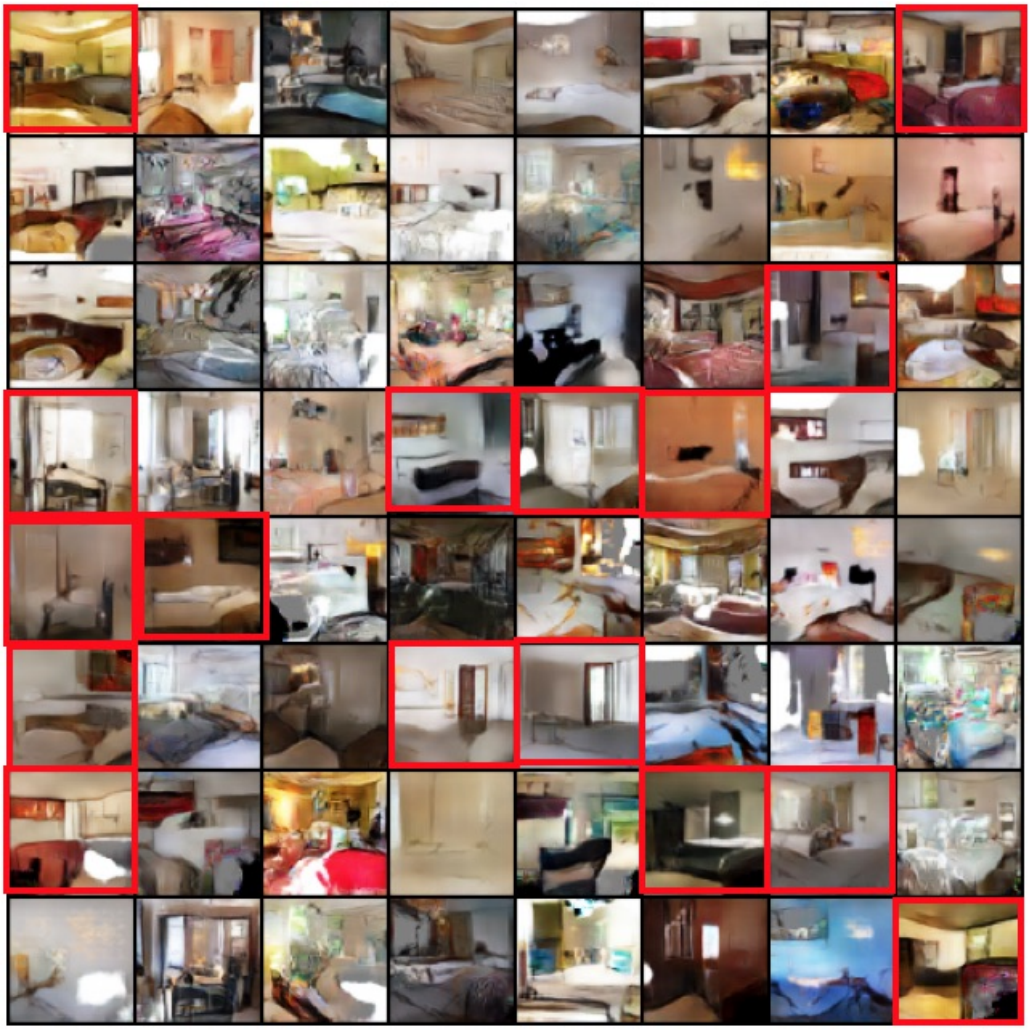

Figure 10. Large version of Figure 4 (d) (WGAN-TS)

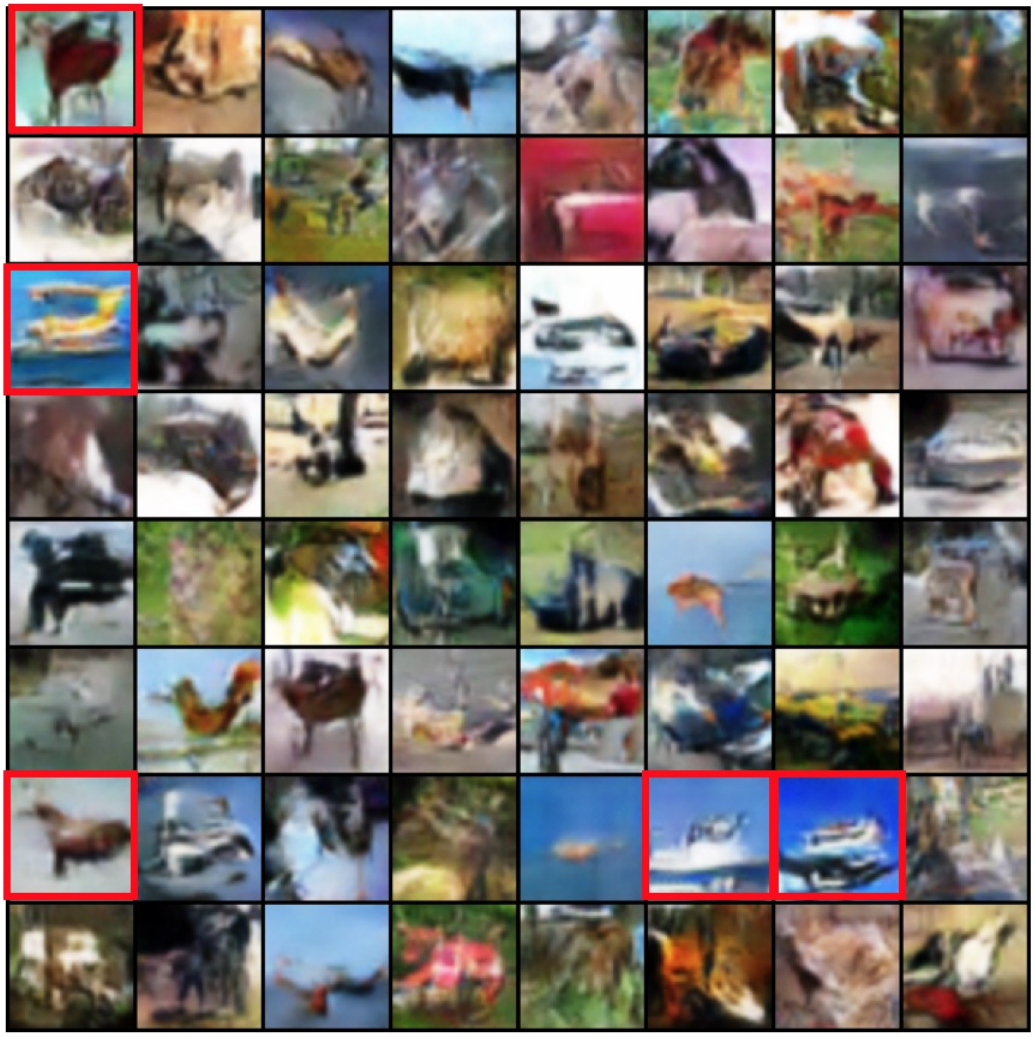

Figure 11. Large version of Figure 6 (a) (WGAN)

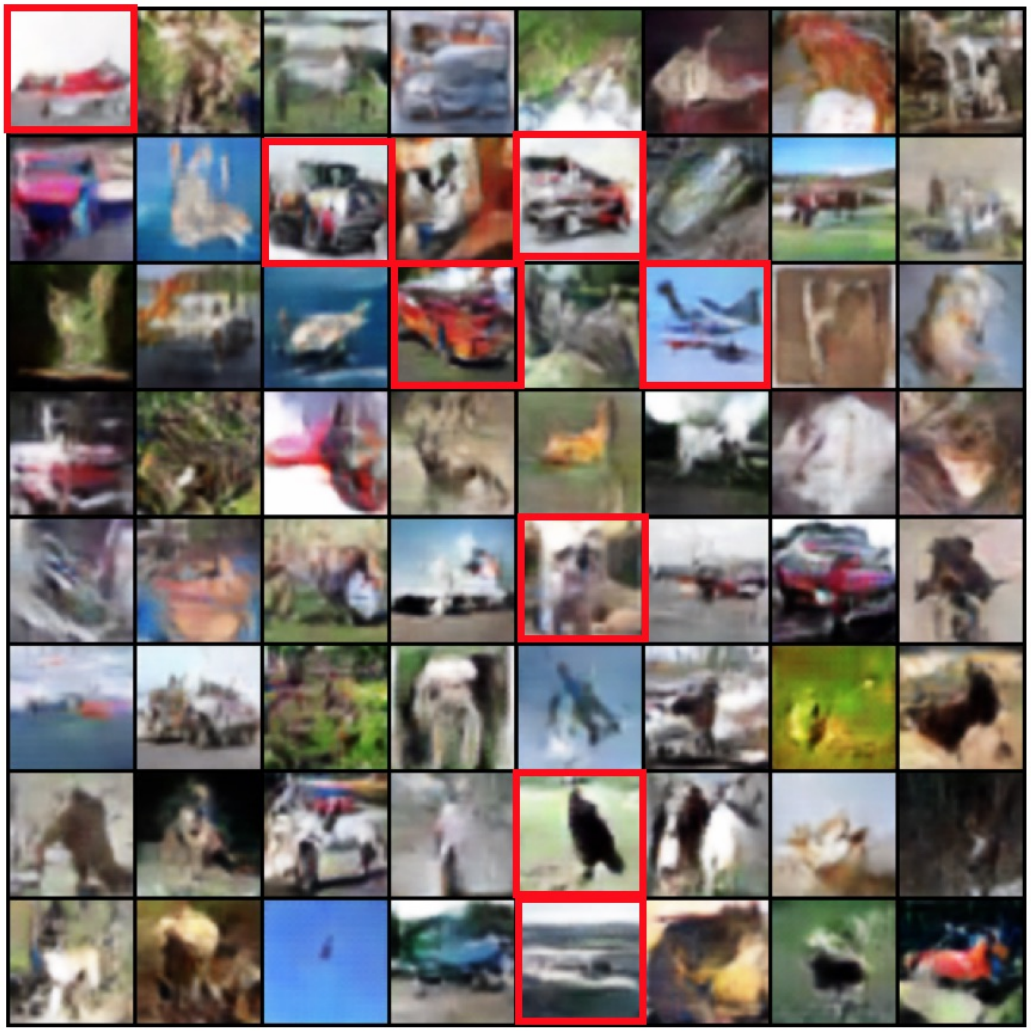

Figure 12. Large version of Figure 6 (b) (WGAN-GP)

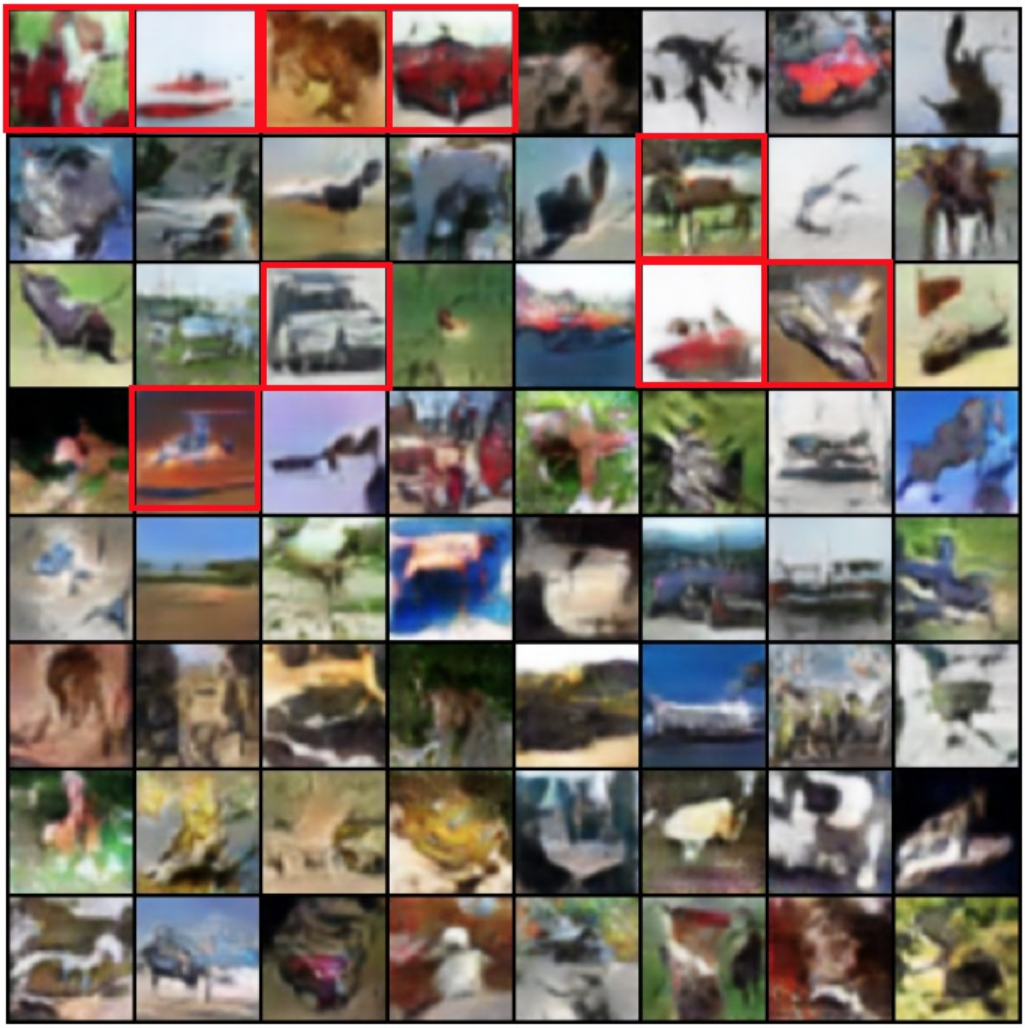

Figure 13. Large version of Figure 6 (c) (SN-WD)

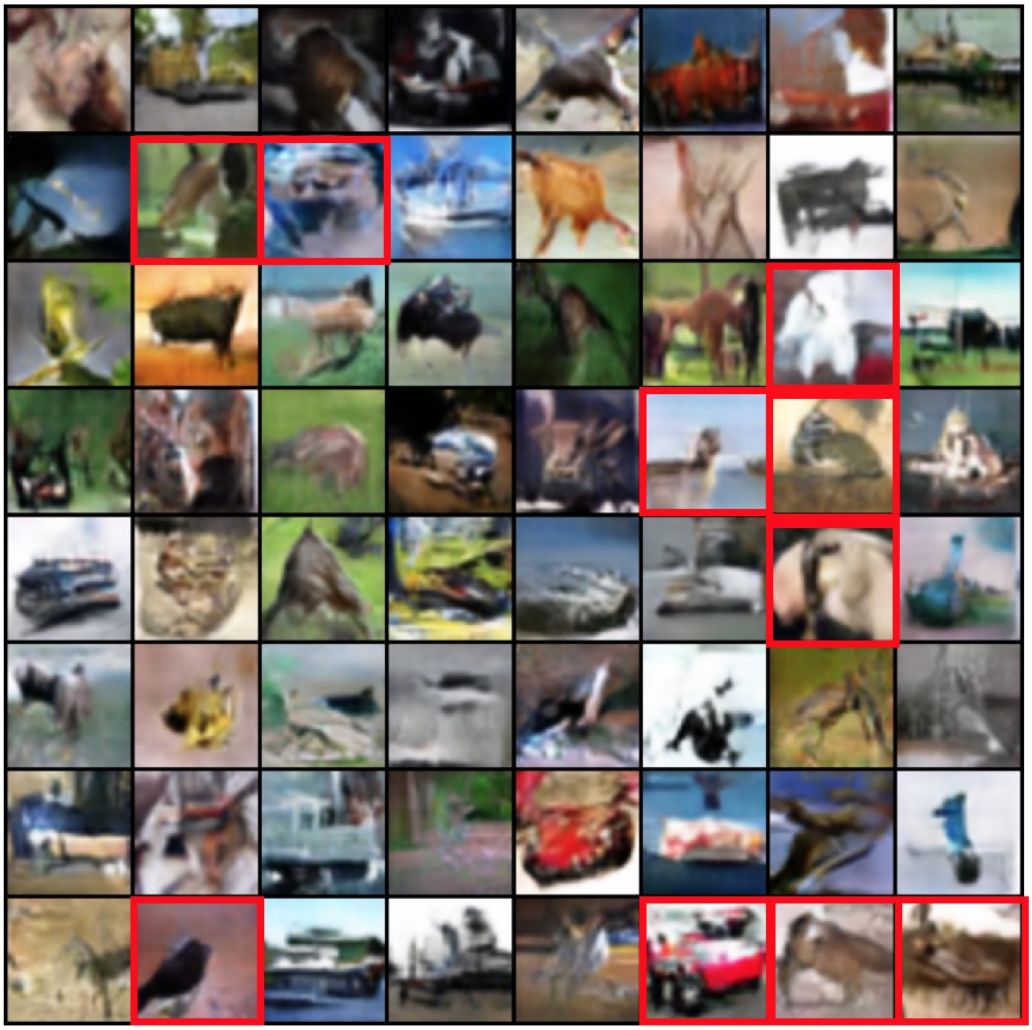

Figure 14. Large version of Figure 6 (d) (WGAN-TS)

Figure 15. First row: images selected in Figure 11 (WGAN). Second row: images selected in Figure 12 (WGAN-GP). Third row: images selected in Figure 13 (SN-WD). Fourth row: images selected in Figure 14 (WGAN-TS). We put the possible category under each image. WGAN generates the least number of recognizable images. WGAN-TS, WGNA-GP and SN-WD generate comparable numbers of recognizable images.