

# GAN を用いた複数の特徴を持つ顔画像の生成

2016SS070 杉崎智仁

指導教員：小市俊悟

## 1 はじめに

本研究では、機械学習の手法を用いて指定された特徴を持った顔画像の作成を行う。顔画像というのは複数の特徴を持つのが普通であり、それらの特徴をすべて持つ顔画像の生成が現在課題となっている。そこで、本研究では、指定された複数の特徴を持つ画像の自動作成を敵対的生成ネットワーク (GAN)[1] と呼ばれる機械学習の手法を用いて実現することを目指し、様々な解決法を試みる。

## 2 手法の説明

### 2.1 GAN

GAN は、生成器と識別器と呼ばれるネットワークが競い合うように学習を行う。その際、学習の進み具合の指標となる損失関数が生成器、識別器にそれぞれ存在する。生成器の損失関数  $L_G$  を式 (1) に、識別器の損失関数  $L_D$  を式 (2) に示す。ただし、 $m$  はバッチサイズ、 $G(z)$  はノイズ  $z$  をもとに生成した贋作データ、 $D(x)$  は識別器の画像  $x$  に対する識別結果 ( $[0,1]$ ) を表す。また、バッチサイズとは、学習する際に異常値の影響を小さくするために、データセットを幾つかのサブセットに分けたそれぞれのサブセットに含まれるデータの個数のことである。

$$L_G = \frac{1}{m} \sum_{i=1}^m (D(G(z_i)) - 1)^2 \quad (1)$$

$$L_D = \frac{1}{m} \sum_{i=1}^m [(D(x_i) - 1)^2 + (D(G(z_i)) - 0)^2] \quad (2)$$

生成器と識別器は損失関数が小さくなるように、つまり、生成器は識別器が本物と識別する ( $D(G(z)) = 1$ ) ように、識別器は訓練データを本物 ( $D(x) = 1$ )、贋作データを偽物と識別できる ( $D(G(z)) = 0$ ) ように学習をする。

## 3 研究の説明

複数の特徴を持つ画像生成をする際の問題点は、二つある。一つは複数の特徴を備える訓練データが少ないこと、もう一つは、複数の特徴を学習しようとするが故に、互いに学習を妨げあってしまう生成器と識別器の学習がスムーズに行えないことである。

### 3.1 解決法 1

奇数 epoch では特徴 1 を持つデータセットを、偶数 epoch では特徴 2 を持つデータセットを用いて特徴を交互に学習させる方法である。ここで、次の手順 1 と 2 を一回実行することを 1epoch と呼んでいる。

- 1 : データセットをバッチサイズに従って  $N$  個のサブセットに分ける。
- 2 : 各サブセットを学習に回す。つまり、 $N$  回学習を繰り返す。

1 と 2 の手順により、データセットに含まれるデータは少なくとも 1 回は学習に用いられることになる。この方法により生成した結果を図 1 に示す。



図 1 解決法 1 による髭が生えており、かつスキンヘッドである人物の顔画像の生成結果

### 3.2 解決法 2

複数ある特徴の各々について識別する識別器をあらかじめ用意しておき、それらの出力結果も利用して、新たな識別器の学習を進めるという方法である。下記は、二つの特徴について学習することを想定した場合である。二つの特徴について学習させるための識別器  $D_1, D_2$ 、最終的に二つの特徴について識別できるように学習をさせる識別器  $D_M$  を用意した。

STEP1: 特徴 1 について、 $D_1$  に識別できるように学習をさせる。

STEP2: 特徴 2 について、 $D_2$  に識別できるように学習をさせる。

STEP3: 特徴 1, 2 について、 $D_M$  に識別できるように学習をさせる。

STEP1, STEP2 における識別器  $D_1, D_2$  の学習方法は、第 2.1 節で紹介した GAN の学習方法と同様である。STEP3 では、新たな識別器  $D_M$  に、二つの特徴を学習させる。この識別器  $D_M$  には、最終的に、指定された二つの特徴を備える画像に対しては値 2 を出力させることを想定している。このとき、STEP3 において、次のような 6 個の関数を利用することで、所望の学習が実現されるのではないかと考えた。

識別器  $D_M$  が訓練データを正しく識別するほど関数値が小さくなる関数

$$L_{D_M}^{(1)} = \frac{1}{m} \sum_{i=1}^m (\min\{D_M(x_i), 1\} - 1)^2 \quad (3)$$

訓練データに対して識別器  $D_M$  には 1 か 2 を出力するようにする。

識別器  $D_M$  が贋作データを正しく識別するほど関数値が小さくなる関数

$$L_{D_M}^{(2)} = \frac{1}{m} \sum_{i=1}^m (D_M(G_M(z_i)) - 0)^2 \quad (4)$$

識別器  $D_M$  が識別器  $D_1 + D_2$  と同じ識別をするほど関数値が小さくなる関数

$$L_{D_M}^{(3)} = \frac{1}{m} \sum_{i=1}^m [D_M(G_M(z_i)) - \{D_1(G_M(z_i)) + D_2(G_M(z_i))\}]^2 \quad (5)$$

$D_1 + D_2$  は二つの特徴を備える画像に対して値 2 を出力すると考える。最終的に  $D_M$  の識別結果は、 $D_1 + D_2$  の識別結果と一致させる。

生成器  $G_M$  が生成した贋作データを識別器  $D_M$  が訓練データと誤認するほど関数値が小さくなる関数 (その 1)

$$L_{G_M}^{(1a)} = \frac{1}{m} \sum_{i=1}^m (D_M(G_M(z_i)) - 1)^2 \quad (6)$$

生成器  $G_M$  が生成した贋作データを識別器  $D_M$  が訓練データと誤認するほど関数値が小さくなる関数 (その 2)

$$L_{G_M}^{(1b)} = \frac{1}{m} \sum_{i=1}^m (D_M(G_M(z_i)) - 2)^2 \quad (7)$$

生成器  $G_M$  と識別器  $D_M$  の学習が適切に進んだとき、 $G_M$  は  $D_M$  が値 2 を出力する画像を生成させる。

生成器  $G_M$  が生成した贋作データに対して識別器  $D_1 + D_2$  が値 2 を出力するほど関数値が小さくなる関数

$$L_{G_M}^{(2)} = \frac{1}{m} \sum_{i=1}^m [\max\{0, D_1(G_M(z_i)) + D_2(G_M(z_i)) - 1\} - 1]^2 \quad (8)$$

値  $D_1(G_M(z_i)) + D_2(G_M(z_i))$  が 2 であれば、値  $\max\{0, D_1(G_M(z_i)) + D_2(G_M(z_i)) - 1\}$  は 1 となる。値  $D_1(G_M(z_i)) + D_2(G_M(z_i))$  が 0 や 1 であれば、値  $\max\{0, D_1(G_M(z_i)) + D_2(G_M(z_i)) - 1\}$  は 0 である。

STEP3 において、最初 100epoch 程度は生成器  $G_M$  がおよそ顔画像を出力できるようにするために解決方法 1 とほぼ同様の学習を行う。

その後、総 epoch の 4 割程度までは、生成器  $G_M$  と識別器  $D_M$  について、次の損失関数  $L_{G_M}$  と  $L_{D_M}$  を用いる。

$\alpha_2$  や  $\beta_3$  を相対的に小さくし、 $G_M$  が特徴の備えた画像を出力できるようになることに重みを置く。

$$L_{G_M} = \alpha_1 L_{G_M}^{(1a)} + \alpha_2 L_{G_M}^{(2)}, \quad (9)$$

$$L_{D_M} = \beta_1 L_{D_M}^{(1)} + \beta_2 L_{D_M}^{(2)} + \beta_3 L_{D_M}^{(3)} \quad (10)$$

ここで、 $\alpha_1 + \alpha_2 = 1, \alpha_1 \geq 0, \alpha_2 \geq 0, \beta_1 + \beta_2 + \beta_3 = 1, \beta_1 \geq 0, \beta_2 \geq 0, \beta_3 \geq 0$  である。

総 epoch の 4 割程度を超えたくらいで、損失関数を次の関数  $L'_{G_M}$  と  $L'_{D_M}$  に切り替え、 $\alpha'_2$  や  $\beta'_3$  を相対的に大きくし、 $G_M$  が二つの特徴をどちらも備えた画像を生成できるように重みを置き、さらに  $D_M$  においても  $D_M$  と  $D_1 + D_2$  の出力結果が一致することに重みを置く。

$$L'_{G_M} = \alpha'_1 L_{G_M}^{(1b)} + \alpha'_2 L_{G_M}^{(2)}, \quad (11)$$

$$L'_{D_M} = \beta'_1 L_{D_M}^{(1)} + \beta'_2 L_{D_M}^{(2)} + \beta'_3 L_{D_M}^{(3)} \quad (12)$$

ここで、 $\alpha'_1 + \alpha'_2 = 1, \alpha'_1 \geq 0, \alpha'_2 \geq 0, \beta'_1 + \beta'_2 + \beta'_3 = 1, \beta'_1 \geq 0, \beta'_2 \geq 0, \beta'_3 \geq 0$  である。図 2 は、

$$(\alpha_1, \alpha_2) = \left( \frac{10.0}{10.1}, \frac{0.1}{10.1} \right) \quad (13)$$

$$(\beta_1, \beta_2, \beta_3) = \left( \frac{5.0}{10.1}, \frac{5.0}{10.1}, \frac{0.1}{10.1} \right) \quad (14)$$

$$(\alpha'_1, \alpha'_2) = \left( \frac{1.0}{6.0}, \frac{5.0}{6.0} \right) \quad (15)$$

$$(\beta'_1, \beta'_2, \beta'_3) = \left( \frac{1}{7.0}, \frac{2.0}{7.0}, \frac{4.0}{7.0} \right) \quad (16)$$

とした場合に学習の結果、 $G_M$  が生成した画像である。

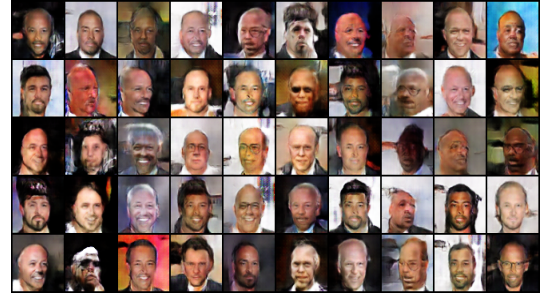


図 2 解決法 2 による髭が生えており、かつスキンヘッドである人物の顔画像の生成結果

## 4 おわりに

複数の特徴を持つ顔画像の生成が 2 つの解決法によりある程度できた。解決法 2 では、重みを変えていくことでより精度の高い画像が生成できる可能性があり、その最適値を見つける方法が必要となる。

## 参考文献

- [1] 宮本圭一郎, 毛利拓也, 大川洋平: 『PyTorch ニューラルネットワーク実装ハンドブック』, 秀和システム, 東京, 2019.