Vol.2017-CVIM-208 No.36
2017/9/16

Conditional Filtered Generative Adversarial Networksを用いた生成的属性制御

金子 卓弘[1,a)]　平松 薫[1,b)]　柏野 邦夫[1,c)]

概要：本稿では生成的属性制御と呼ぶ新しい問題に取り組む．生成的属性制御では，画像の生成または編集を，属性内多様性（例えば，笑顔属性であれば微笑み，大笑い，にやり笑いなどの様々な笑い方）を直感的に制御しながら行えるようにすることを目指す．これを実現するためには，画像の表現空間があった時に，(1) 個人性と属性が分離され，さらに，属性に対して (2) 高い表現力と (3) 高い操作性が得られていることが必要になる．これらを満たすために，本稿では Conditional Filtered Generative Adversarial Networks（CFGAN）と呼ぶ Conditional GAN（CGAN）の新しい拡張モデルを提案する．CGAN は GAN を条件付き設定に拡張したもので，属性の観測変数を生成器と識別器の入力に組み込むことで，表現空間内で個人性と属性を分離することを可能にしている．一方で，表現力と操作性は観測変数に強く制約されており，例えば，観測変数が属性の有無を表すバイナリであればオン・オフの制御しかできなかった．これに対して，CFGAN では新たにフィルタリング構造と多次元の隠れ変数を導入し，属性の観測変数の値に応じて隠れ変数のフィルタリングを行う．これにより属性は多次元的に表現されるため表現力を高めることが可能であり，さらに，フィルタリング構造と隠れ変数の分布形状を工夫することで様々な制御を実現することが可能である．実験では，CFGAN を MNIST，CUB，CelebA データセットに適用し，様々なデータに対して属性内多様性を制御しながら画像を生成または編集できることを示す．さらに，本手法を属性転写と属性に基づく画像検索の二つのタスクに適用し，本手法が属性の表現学習にも有用であることを示す．

1. はじめに

コンピュータビジョンや機械学習の分野では，画像を構成する「因子」を明らかにすることは根源的な問題の一つであり，そのキーとなる技術として生成モデルの研究が古くから行われている．特に，コンパクトでかつ表現力のある画像の表現空間を得ることは長年未解決問題であったが，近年の深層学習に基づく生成モデル [1], [2], [3] の著しい発展により解決の糸口が見え始めている．これらの研究によって得られた表現空間はコンパクトであるだけではなく，空間内でランダムに選択した値からリアリティのある画像を生成することも可能であり，高い表現力も持つ．一方で，この表現空間内では各次元が複雑に関係しあっており，個々は特別な意味を持っていないため，思い通りの画像を生成することは簡単ではないという課題があった．

この課題を克服するために，本研究では表現空間の「操作性」に新たに着目する．特に，画像の属性に対する操作性に着目し，画像の属性内多様性（例えば，笑顔属性であれば微笑み，大笑い，にやり笑いなどの様々な笑い方）を直感的に制御しながら画像を生成または編集できるようにすることを目指す．本稿では，この問題を「生成的属性制御」と呼ぶ．提案モデルを用いた生成的属性制御の例を図

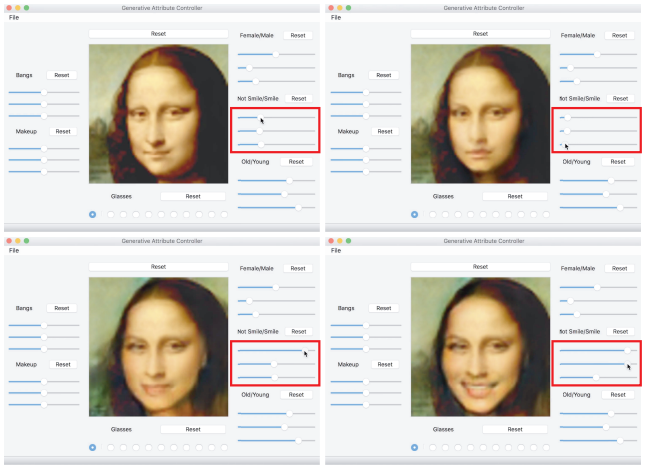


図 1: 表情属性に対する生成的属性制御の例．本システムでは表情は多次元的に表されており（赤枠部，3 次元のスライドバーで表現），スライドバーを動かすことで様々な表情の顔画像に直感的に変更可能．

[1] NTT コミュニケーション科学基礎研究所
3-1, Morinosato Wakamiya, Atsugi-shi, Kanagawa 243–0198, Japan
[a)] kaneko.takuhiro@lab.ntt.co.jp
[b)] hiramatsu.kaoru@lab.ntt.co.jp
[c)] kashino.kunio@lab.ntt.co.jp

1 に示す．

生成的属性制御を実現するためには，以下の三つを満たす画像の表現空間を得ることが必要である．一つ目が「個人性と属性の分離」である．本研究では属性だけを変えることが目的であり，そのためには個人性と属性が分離されていることが必要である．二つ目が「属性に対する高い表現力」である．例えば，「笑顔」と一口に言っても微笑み，大笑い，にやり笑いなど様々なものがあり，そのような属性の多様性に対して十分な表現力を持っていることが必要である．三つ目が「属性に対する高い操作性」であり，人間が直感的に制御できるようにするために表現空間内で属性がうまく整理されていることが必要である．

これら三つの要求事項を満たすために，本稿では，Conditional Filtered Generative Adversarial Networks（CFGAN）と呼ぶ Conditional GAN（CGAN）[4] の新しい拡張モデルを提案する．CGAN は GAN [1] を条件付き設定に拡張したものであり，属性の観測変数を生成器と識別器の入力に組み込むことで，表現空間内で個人性と属性を分離することを可能にしている．一方で，表現力と操作性は観測変数に強く制約されており，例えば，観測変数が属性の有無を表すバイナリ（例，笑顔の有無）であればオン・オフ（例，笑顔か笑顔でないか）の制御しかできず，属性内多様性（例，どんな笑顔か）を扱うことはできなかった．この課題を解決するために，CFGAN では新たにフィルタリング構造と多次元の隠れ変数を導入し，属性の観測変数の値に応じて隠れ変数のフィルタリングを行う．これにより属性は多次元的に表現されるため表現力を高めることが可能である．さらに，フィルタリング構造と隠れ変数の分布形状を工夫することで様々な制御，具体的には，典型的な GUI であるスライドバーやラジオボタンによる制御を実現することが可能である．実験では，CFGAN を文字，鳥，顔画像のデータセットに適用し，様々なデータに対して属性内多様性を制御しながら画像を生成または編集できることを示す．さらに，本手法を属性転写と属性に基づく画像検索の二つのタスクに適用し，本手法が属性の表現学習にも有用であることを示す．

貢献：本研究の貢献をまとめると以下である．(1) **生成的属性制御**という新しい問題を提起．(2) 個人性と属性が分離され，さらに，高い表現力と操作性の持つ表現空間を学習するために **CFGAN** と呼ぶ CGAN の新しい拡張モデルを提案．(3) 実験では，様々なデータに対して属性内多様性を制御しながら画像を**生成**または**編集**できることを実証．さらに，属性**転写**と属性に基づく画像**検索**に適用し，本手法が属性の表現学習にも有用であることを実証．

2. 関連研究

画像編集：画像編集はコンピュータグラフィックスの分野を中心に熱心に研究されており，様々なタスクが取り組まれている．属性に基づく画像編集としては，用例に基づく方法 [5], [6], [7], [8] やモデルに基づく方法 [9], [10] が代表的な手法である．前者は参照画像のパッチを対象画像に転写することで属性の編集を行うものであるが，この方法は画像に対して強い制約（例えば，正面画像のみ [5]）がある点に限界があった．一方，後者は物体のモデルを構築し，そのモデルに基づき属性を編集するものであるが，この手法は特定のタスク（例えば，顔の正面化 [9]）に限定されたものであり，任意の属性に適用することは難しかった．これら限界の一つの要因としては，画像を低レベルな表現空間で編集している点が挙げられる．これに対して，本研究では深層学習に基づく生成モデルを用いることで，よりセマンティックな領域で画像を編集することを可能にしており，様々な条件の画像に対して様々な属性の編集をすることを可能にしている．

深層学習に基づく生成モデル：近年の深層学習の発展は著しいが，生成モデルにおいても深層学習を組み込むことで大きな革新が起きつつある．特に，深層学習に基づく確率的な生成モデルである GAN [1]，Variational Autoencoder（VAE）[2], [3]，Autoregressive Model [11] は，乱数から本物と区別のつかない画像を生成することを可能にし，注目を集めている．これらの研究の中には，本研究と同様に画像生成における属性の操作性に着目し，観測変数をモデルに取り組んだもの [4], [12], [13], [14], [15] があるが，これらの属性表現は観測変数に強く依存しており，属性内多様性に対応するためには豊富な教師データを得ることが必要であった．本研究と同様に属性を隠れ変数で表現するものとしては，Deep Convolutional Inverse Graphics Network [16] や Adversarial Autoencoder [17] があるが，前者はグラフィックエンジンを用いて学習データを生成することが必要なため自然画像への適用は難しく，後者は学習方法が確立されておらず適用できるデータが比較的な簡単なデータに限られるという点が課題にあった．これに対して，CFGAN は GAN の自然な拡張モデルであり，近年提案された様々な学習の工夫 [18], [19] を取り入れることができ，比較的安定的に学習することが可能である．

属性表現：属性をどう表現するかということはコンピュータビジョン分野で熱心に議論されていることであり，例えば，属性の有無をバイナリで表す方法 [20], [21], [22] がある．この方法では，多様な属性があった時に表現に限界があるため，相対的な関係で表す方法 [23]，さらに，順位づけには限界があるため識別的かどうかという基準で表す方法 [24] などが提案されている．これらの研究では属性表現の複雑さと難しさを指摘しており，人手で全てのルールを定義することは困難であることを示唆している．これに対して，本研究では解釈可能な属性表現を限られた観測変数だけから自動的に学習，発見できるものであり，属性表現の問題に解決の糸口を与えることができると考えている．

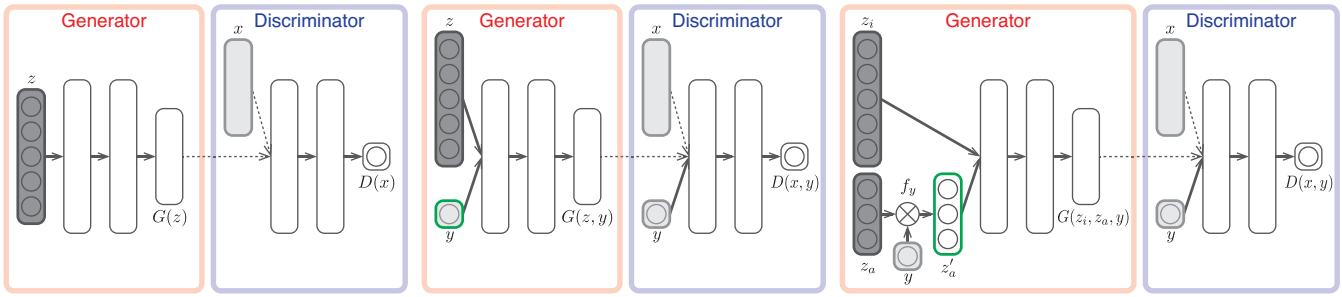


図 2: 従来手法（GAN，CGAN）と提案手法（CFGAN）のネットワーク構造の差異．薄いグレーは観測変数，濃いグレーは隠れ変数，緑枠は属性の制御に使用可能な変数を表す．(a) GAN では属性は明示的に表されていないため属性に関して制御不可である．(b) CGAN では属性は観測変数 y で表現され制御可能であるが，表現力は観測変数に制約される．例えば，観測変数が属性の有無を表すバイナリであればオン・オフの制御しかできない．(c) CFGAN では属性は多次元の隠れ変数 z'_a で表現され，多次元的に，つまり，より表現力の高い空間で制御可能である．

3. 提案手法

3.1 GAN・CGAN

本節では，提案手法である CFGAN のベースである GAN [1] および CGAN [4] について説明する．GAN は生成モデルを Min-Max 最適化を用いて学習するものであり，目的は真のデータ分布 $P_{\text{data}}(x)$ に一致するような生成分布 $P_G(x)$ を学習することである．GAN は二つのネットワークで構成されており，一つは生成器 G で乱数 $z \sim P_z(z)$ をデータ空間 $x = G(z)$ に写像する．もう一つは識別器 D で x が P_{data} からサンプリングされたものであれば確率 $p = D(x) \in [0, 1]$ を付与し，P_G からサンプリングされたものであれば確率 $1 - p$ を付与する．D と G は以下の Min-Max の目的関数で最適化が行われる．

$$\min_G \max_D \mathbb{E}_{x \sim P_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z \sim P_z(z)}[\log(1 - D(G(z)))]. \quad (1)$$

この目的関数では D は真の画像と生成画像の良い識別境界を見つけるように最適化が行われ，同時に G は真のデータ分布に近づくように最適化が行われる．このモデルでは図 2(a) にあるように属性は明示的に表現されていないため，属性を制御しながら画像を生成することはできない．

CGAN は GAN を条件付き設定に拡張したものであり，G と D は属性の観測変数 y を入力に持つ．目的関数は以下のようになる．

$$\min_G \max_D \mathbb{E}_{x,y \sim P_{\text{data}}(x,y)}[\log D(x, y)] + \mathbb{E}_{z \sim P_z(z), y \sim P_y(y)}[\log(1 - D(G(z, y), y))]. \quad (2)$$

条件付きの生成器と識別器を Min-Max 最適化をすることで，表現空間（生成器の入力空間）内で表現の分離，つまり，z と y がそれぞれ個人性と属性を表現するようにすることが可能である．これにより，y を操作することで属性を制御しながら画像を生成することが可能である．CGAN のネットワーク構造を図 2(b) に示す．

3.2 CFGAN

前述したように CGAN では属性を制御しながら画像を生成することが可能であるが，その表現力は観測変数 y に制約されており，例えば，y が属性の有無を表すバイナリであればオン・オフの制御しかできない．仮に属性に関して詳細な教師データを得ることができれば，より高い表現力を得ることができるが，第 2 章「属性表現」で述べたように，そもそも属性について詳細なレベルまで人手で定義することは簡単ではない．

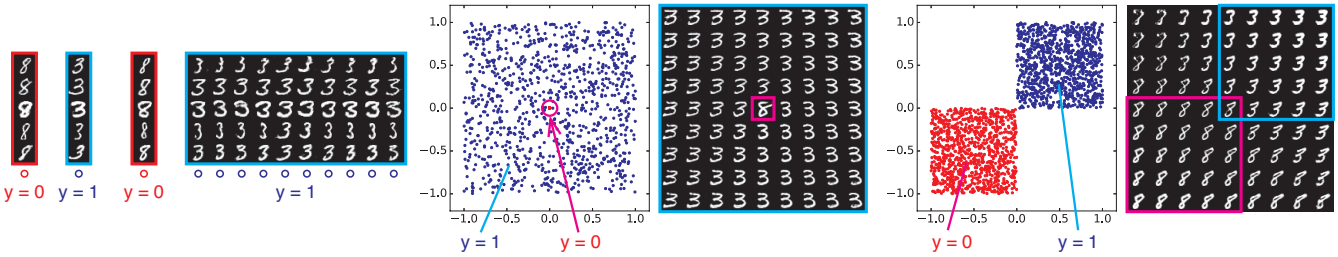
この課題を克服するために，本研究は詳細な教師データを要することなく，属性の表現能力を向上させられる方法（CFGAN）を提案する．具体的には，以下のようなフィルタリング構造を生成器の入力部分に導入する．CFGAN のネットワーク構造を図 2(c) に示す．

$$z'_a = f_y(z_a). \quad (3)$$

ここで，$z_a \sim P_{z_a}(z_a)$ は乱数である．これを関数 f_y を用いてフィルタリングをすることで，y の値に応じて空間内で領域分けされた属性の隠れ変数 z'_a を得ることを可能にする．CFGAN の目的関数は以下のようになる．

$$\min_G \max_D \mathbb{E}_{x,y \sim P_{\text{data}}(x,y)}[\log D(x, y)] + \mathbb{E}_{z_i \sim P_{z_i}(z_i), z_a \sim P_{z_a}(z_a), y \sim P_y(y)} [\log(1 - D(G(z_i, z_a, y), y))]. \quad (4)$$

ここで，z_i は CGAN の z に相当するものであり，CGAN と同様に条件付きの設定で Min-Max 最適化をすることで表現の分離，つまり，z_i と z'_a がそれぞれ個人性と属性を表現するようにすることが可能である．この際，z_a に多次元の変数を用いることで属性を多次元的に表現，つまり，より表現豊かに表現することが可能である．さらに，詳細については次章で述べるが，f_y と z_a の構成を工夫することによって，様々な制御を実現することが可能である．



(a) CGAN (b) CFGAN + RB (c) CFGAN + SB-I (d) CFGAN + SB-II

図 3: CGAN，CFGAN + RB，CFGAN + SB-I，CFGAN + SB-II を用いた時の属性の制御例．ここでは，数字の「3」と「8」のみを含んだデータを用いており，「3」が属性のある時（$y = 1$），「8」が属性のない時（$y = 0$）としている．(a) 各行は同じ z，異なる y から生成された例を表す．CGAN では属性のオン・オフの制御だけが可能である．(b) 各行は同じ z_i，異なる z'_a から生成された例を表す．CFGAN + RB では $y = 1$ の状態に対して離散的に制御することが可能である．(c) CFGAN + SB-I では，左図のように中心だけが $y = 0$ に対応して他は $y = 1$ に対応するような z'_a 空間を得ることが可能であり，この空間で値を変化させると，右図のように中心に近づくにつれて「8」（$y = 0$）の度合いが増すように連続的に変化させることが可能である．(d) CFGAN + SB-II では，左図のように負の領域は $y = 0$ に対応して正の領域は $y = 1$ に対応するような z'_a 空間を得ることが可能であり，この空間で値を変化させると，右図のように「8」（$y = 0$）から「3」（$y = 1$）までを連続的に変化させることが可能である．

4. 制御のための工夫

4.1 フィルタリング構造

CFGAN ではフィルタリング構造を工夫することで様々な制御方法を実現することができるが，ここでは代表例として典型的な GUI のコントローラーであるラジオボタン制御とスライドバー制御の実現方法について説明する．

ラジオボタン制御：ラジオボタンは複数の選択肢の中から一つだけ選択できる GUI であり，属性を離散的に制御することを可能にする．ラジオボタン制御（**RB**）は f_y と z_a を以下のように構成することで実現することができる．

$$f_y(z_a) = \begin{cases} z_a & (y = 1) \\ 0 & (y = 0) \end{cases}, z_a \sim \mathrm{Cat}\left(K = k, p = \frac{1}{k}\right). \tag{5}$$

ここで，z_a はカテゴリ数 k のカテゴリカル分布からサンプリングされ，f_y は属性が存在する時（$y = 1$）はそのまま z_a を用い，属性が存在しない時（$y = 0$）はゼロにする処理を行う．CFGAN + RB を用いた時の属性の制御例を図 3(b) に示す．

スライドバー制御：スライドバーはバーを動かすことによって連続的に値を変えられる GUI であり，属性を連続的に制御することを可能にする．スライドバー制御の実現方法は二通りあり，一つ目の方法（**SB-I**）が，属性がない時を中心として連続的に変化させる方法で，これは f_y と z_a を以下のように構成することで実現することができる．

$$f_y(z_a) = \begin{cases} z_a & (y = 1) \\ 0 & (y = 0) \end{cases}, z_a \sim \mathrm{Unif}\,(-1, 1). \tag{6}$$

ここで，z_a は一様分布からサンプリングされ，f_y は RB と同様の処理を行う．CFGAN + SB-I を用いた時の属性の制御例を図 3(c) に示す．

もう一つの方法（**SB-II**）は，属性がない時とある時を連続的に変化させる方法で，これは f_y と z_a を以下のように構成することで実現することができる．

$$f_y(z_a) = \begin{cases} |z_a| & (y = 1) \\ -|z_a| & (y = 0) \end{cases}, z_a \sim \mathrm{Unif}\,(-1, 1). \tag{7}$$

ここで，z_a は SB-I と同様であるが，f_y が $y = 0$ の時は負の領域に写像し，$y = 1$ の時は正の領域に写像する点が異なる．CFGAN + SB-II を用いた時の属性の制御例を図 3(d) に示す．

CGAN と CFGAN の関係：CFGAN + RB においてカテゴリ数 $k = 1$，つまり，ラジオボタンの選択が可能でないとすると，CFGAN は CGAN と一致し，CGAN は CFGAN の特殊な例であると言える．このことから，CFGAN は CGAN の自然な拡張であると言える．

4.2 学習方法

前述したように CFGAN では CGAN と同様に条件付き設定で G と D を学習することで，z_i と z'_a 間で個人性と属性の表現の分離が可能であるが，ナイーブな学習方法では，z'_a の各次元同士の関係に対しては何も制約を与えられず，独立した概念を表すようにはできない．これは，z'_a を用いて属性を制御しようとした時に好ましくない性質である．

この課題を解決するために，本研究では情報理論に基づく正則化 [19] を条件付き設定に拡張したものを CFGAN の学習の際に用いる．これは z'_a と $G(z_i, z_a, y)$ 間の条件付き相互情報量 $I(z'_a; G(z_i, z_a, y)|y)$ の最大化を行うものであり，z'_a の各次元が独立した概念を表すように制約を与える．なお，この情報量を直接計算しようとすると事後分布 $P(z'_a|x, y)$ を求める必要があるが，これは計算困難である．そこで，実際には，補助分布 $Q(z'_a|x, y)$ を導入して $P(z'_a|x, y)$ を近似し，下限の最大化を行う．

図 4: 画像編集の例．(1) 入力画像．(2) エンコーダーを用いて再構成した後の画像．(3) 勾配法を用いて再構成した後の画像．(4) 属性（髪型）を変更後の画像．(5) ポストプロセス後の画像．

5. 画像編集のための工夫

5.1 隠れ変数の推定

CFGAN は画像生成のためのモデルであり，属性の制御は G の入力空間で行われる．そのため，CFGAN を画像編集に用いるためには，画像 x が与えられた時に隠れ変数 z_i と z'_a を推定することが必要である．この問題に対して，本研究では段階的な推定方法，具体的には，まず x から y を推定し，x と y から z'_a を推定し，最後に z'_a と x から z_i を推定するという方法を用いる．

x から y の推定は以下の式によって行うことができる．

$$y^* = \arg\max_y P(y|x). \tag{8}$$

これは標準的な識別タスクであり，これを解くために本研究では分類器 $C(x)$ を別途学習している．y を得ることができたら，第 4.2 節で述べた $Q(z'_a|x,y)$ を用いることで，z'_a を推定することが可能である．

z'_a と x から z_i を推定する際は Manifold Projection 法 [25] を用いて以下の式の最適化を行う．

$$z_i^* = \arg\min_{z_i} \mathcal{L}(G(z_i, z'_a), x). \tag{9}$$

ここでは分かりやすさのため $G(z_i, z_a, y)$ を $G(z_i, z'_a)$ と書き換えている．ここで，$\mathcal{L}(x_1, x_2)$ は二つの画像間の距離を測る関数であるが，実験では，近年の Perceptual Loss [26], [27] の成功に従い，画像空間と D の中間層から抽出された特徴量空間の両空間で距離を計測するよう設計している．また，式 (9) は微分可能なニューラルネットワークで構成されているため，直に勾配法を用いて最適化を行うこともできるが，非凸な問題であるため計算効率は初期値に依存する．そこで，本研究では，x から z_i を推定するエンコーダー $E(x)$ を別途用意する．このエンコーダーは目的関数 $\mathcal{L}(G(E(x), z'_a), x)$ を用いて学習することができる．学習した $E(x)$ を用いて z_i を推定することで良い初期値を求め，それから勾配法を用いることで最適な解を効率的に求められるようにする．図 4(1)(2)(3) に入力画像，エンコーダー，勾配法を用いて再構成した後の画像の例を示す．この結果からは，二段階の最適化を行うことで，細部（例えば眉毛など）の再構成が改善できていることが分かる．

5.2 ポストプロセス

上述した再構成のプロセスでは，高次元の画像を一度低次元空間に圧縮し復元するという処理を行うため，ピクセルレベルで完全に一致した画像を再構成することは困難である．しかし，この性質は CFGAN を画像編集に用いようとした時に好ましくない性質である．

そこで，この影響を緩和するため，本研究ではポストプロセスを用いる．具体的には，従来の Masking 手法 [28] を拡張したものを用いる．

$$\begin{aligned} \tilde{x} &= x_{\mathrm{rec}} + M\Delta + (1-M)\Delta' \\ \Delta &= x_{\mathrm{mod}} - x_{\mathrm{rec}}, \Delta' = x - x_{\mathrm{rec}}. \end{aligned} \tag{10}$$

上式で，x は入力画像，x_{rec} が再構成後の画像，x_{mod} が属性変更後の画像である．マスク M は Δ と Δ' の混合の割合を決めるものであり，従来研究 [28] では，Δ の絶対値のチャンネル毎の平均を求めた後，Standard Gaussian Filter を用いて平滑化し，0 と 1 の間に値を収めるという処理を行うことで，ノイズに頑健な処理を行えるようにしていた．本研究では，試行錯誤の結果 Standard Gaussian Filter ではなく，Scaled Gaussian Filter，$\alpha \cdot g(\cdot)$ を用いた方が良いことが分かったため以下のマスクを用いる．

$$M = \min(\alpha \cdot g(|\overline{\Delta}|; \sigma), 1). \tag{11}$$

ここで α はスケールパラメータであり，どのぐらい小さな変化まで許容するかの尺度を表し，σ は Gaussian カーネルの標準偏差で平滑化の度合いを表す．このポストプロセスは簡単なものであるため，α と σ の最適な値はインタラクティブに操作しながら探すことも可能である．図 4(4)(5) にポストプロセス前後の画像の例を示す．この結果より，ポストプロセスを用いることで再構成誤差を緩和し，個人性をより保持しながら属性を変化させられることが分かる．

6. 評価実験

評価実験では，提案手法を (1) 属性に基づく画像**生成**，(2) 属性に基づく画像**編集**，(3) 属性**転写**，(4) 属性に基づく画像**検索**の四つのタスクに適用し有効性を検証した．個々の実験結果について述べる前に，まず実験設定について述べる．なお，実験の詳細な説明やより多くの実験結果はプロジェクトページ [*1] に掲載があるので参照されたい．

データセット：実験では，様々なドメインのデータセット，具体的には，手書き文字画像（MNIST [29]），鳥画像（CUB [30]），顔画像（CelebA [31]）を用いて評価を行った．MNIST データセットは 60,000 枚の訓練画像，10,000 枚のテスト画像を含み，実験では CFGAN の基本的な性質を明らかにするために用いた．CUB データセットは 200 種類の鳥画像から構成されており，6,000 枚の訓練画像，6,000 枚のテスト画像を含む．本実験では，画像の切り抜きが行われたもの [13] を用い，64×64 にリスケールを

[*1] http://www.kecl.ntt.co.jp/people/kaneko.takuhiro/projects/gac/index.html

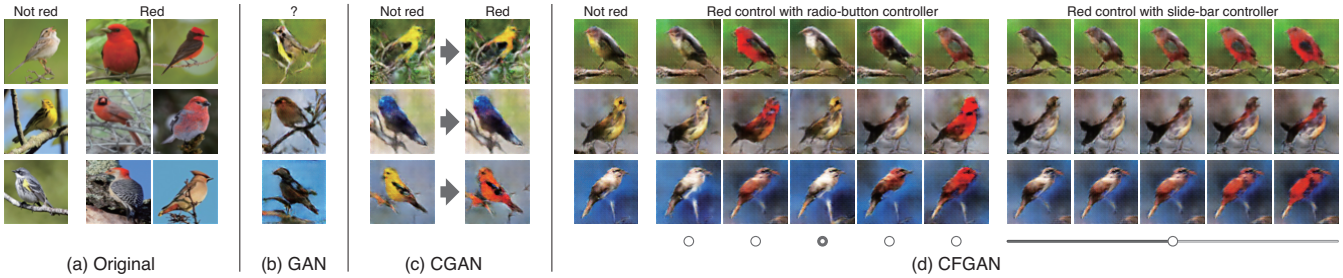


図 5: 赤い鳥生成の例．(c)(d) では，各行は同じ個人性を表す変数から生成し，各列は同じ属性を表す変数から生成した結果を表す．

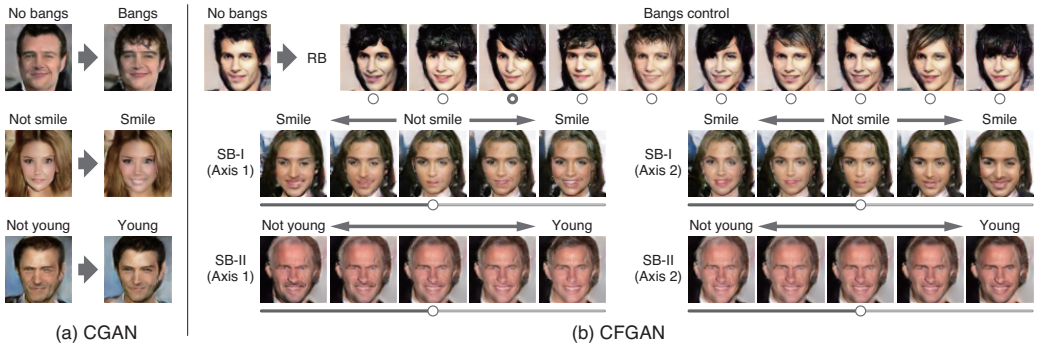


図 6: 顔属性制御に基づく顔画像生成の例．各行は個人性を表す変数は固定して属性を表す変数を変えた時の生成結果を表す．

行った．CelebA データセットは 180,000 枚の訓練画像と 20,000 枚のテスト画像を含み，本実験では位置合わせと画像の切り抜きが行われたものを用い，64×64 にリスケールを行った．なお，過学習を防ぐために CUB データセットと CelebA データセットを用いるときは一般的なデータ拡張の方法 [32], [33] を用いた．

実装詳細：CFGAN の学習を安定化させるために，従来手法（DCGAN [18] および InfoGAN [19]）の学習テクニックを参考にネットワーク構造や学習方法の設計を行った．具体的には，ネットワークは主に Convolution または Deconvolution 層を用いて構成し，活性化関数としては G に対しては ReLU [34]，D に対しては Leaky ReLU [35], [36] を用い，さらに，各層で Batch Normalization [37] を用いて正規化を行った．第 4.2 節で述べた Q は，InfoGAN に従い，ニューラルネットワークで構成した．特に，D と Q は Convolution 層を共有化することにより，計算コストの削減を行った．ラジオボタン制御の際は Q の分布は Softmax 関数を用いて表現し，スライドバー制御の際は Q の分布は Factored Gaussian 関数を用いて表現した．C と E には最終層以外は D と同じ構造を持ったネットワークを用い，特に，C と E は Convolution 層を共有化することにより，計算コストの削減を行った．最適化には Adam [38] を用い，バッチサイズは 128，学習率は D/Q に対しては 2e-4，G に対しては 1e-3，C/E に対しては 1e-3 とし，モーメンタム項 β_1 は 0.5 とした．

比較手法：比較手法としては，属性を制御できない GAN と属性を一次元で表現する CGAN を用いた．なお，深層学習に基づく生成モデルのうち属性を制御可能なものとしては Conditional VAE（CVAE）[13] や VAE/GAN + Visual Attribute Vector [39] などがあるが，これらは属性を一次元のベクトルで表現しているため表現力や操作性は CGAN と同じである．また，CGAN と InfoGAN を愚直に組み合わせた方法も考えられるが，この方法で得られる操作可能な隠れ変数は属性と独立したものであるため，属性に対する表現力は CGAN と変わらないことに留意されたい．

6.1 属性に基づく画像生成

二つの数字間の制御：CFGAN の基本的な性質を明らかにするために，MNIST の一部のデータを使って実験を行った．具体的には，二つの数字を選択し，一つを属性のある状態（$y = 1$），もう一つを属性のない状態（$y = 0$）とした．「3」を属性のある状態，「8」を属性のない状態とした時の実験結果を図 3 に示す．この結果から，CFGAN では属性内多様性が表現できるだけではなく，f_y と z_a の構成を変えることによって様々な制御が可能であることが分かる．

赤い鳥生成：より多様性のあるデータに対する CFGAN の有効性を検証するために CUB データセットを用いて実験を行った．ここでは，312 種類の属性アノテーションのうち「赤」を含むもの（例，赤い羽根，赤い眼，赤い脚など）全てを赤い鳥とみなし，赤い鳥を属性のある状態（$y = 1$），赤くない鳥を属性のない状態（$y = 0$）とした．図 5(a) に示すように世の中には多種多様な赤い鳥が存在することに留意されたい．本実験では，CFGAN としては 5 次元の RB と 1 次元の SB-I を組み合わせたものを用いた．生成結果を図 5 に示す．この結果からは，CGAN では鳥を赤くできるだけでどう赤くするかまでは制御できないのに対し，CFGAN では離散的にも連続的にも制御可能であり，様々な方法で鳥を赤くすることができることが分かる．

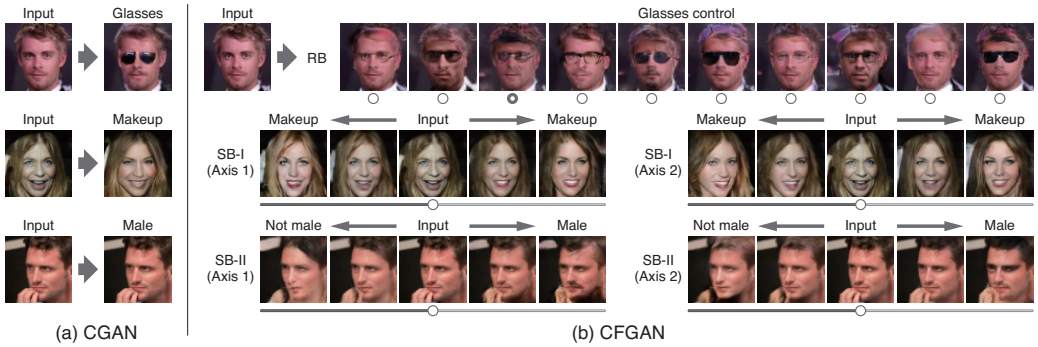


図 7: 顔属性制御に基づく顔画像編集の例．各行は個人性を表す変数は固定して属性を表す変数を変えた時の生成結果を表す．

図 8: 前髪に基づく属性転写の例．対象画像から z_i を抽出し，参照画像から z'_a を抽出し，これらを組み合わせて生成した結果を示す．

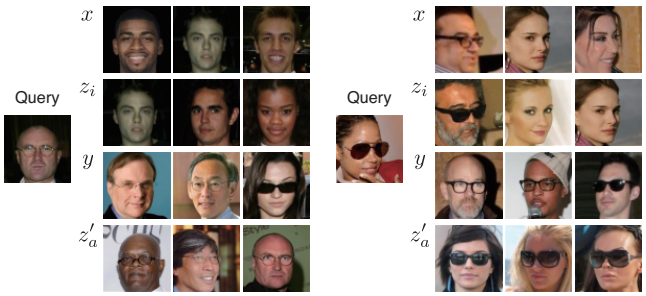


図 9: 眼鏡に基づく画像検索の例．トップ 3 の検索結果を左から順に示す．上から順に x，z_i，y，z'_a の空間で検索した結果を表す．

顔属性の制御：様々な属性に対する CFGAN の有効性を検証するために，CelebA データセットの六つの属性（前髪，眼鏡，化粧，男性，笑顔，若さ）各々に対して三つの制御（10 次元の RB，3 次元の SB-I，3 次元の SB-II）を実装し，実験を行った．スペース上の関係で本稿では抜粋したものを図 6 に示す．この結果からは，顔画像の様々な属性に対して，CGAN ではオン・オフの制御しかできないが，CFGAN では様々な制御が可能であることが分かる．

6.2 属性に基づく画像編集

CFGAN の画像編集に対する有効性を検証するために，前節で述べたモデルと同じモデルを使って画像編集を行った．画像の編集結果を図 7 に示す．なお，本実験では全て第 5.2 節で述べたポストプロセスの適用後の結果を示す．結果からは，CGAN と CFGAN ともに個人性を保持しながら属性が変更できる点で共通する一方で，属性の操作性は画像生成の時と同様に大きく異なることが分かる．

6.3 属性転写

CFGAN によって得られた属性表現の有用性について評価を行うために，本手法を属性転写と属性に基づく画像検索の二つのタスクに適用し，検証を行った．本節では前者について述べ，後者については次節で説明する．属性転写を行うためには，対象画像と参照画像の各々から z_i と z'_a を抽出し，それらを組み合わせて画像を生成すればよい．属性転写の結果を図 8 に示す．ここでは前髪の転写を行っており，CFGAN としては 3 次元の SB-I を用いた．結果からは性別，年齢，顔の向きによらず前髪の詳細なタイプ（髪の分け方など）が転写できていることが分かる．なお，属性を転写しようとしたものは従来研究 [18], [39] でもあったが，これらでは表現空間内で属性が分離できていなかったため，属性表現を得るためには多数のサンプルを集め平均をとることが必要であった．これに対し，提案手法では表現空間内で属性が分離できているため，1 枚のサンプルだけを用いて転写できることに留意されたい．

6.4 属性に基づく画像検索

画像検索に対する有効性を検証するために，距離を測る空間を変えて比較を行った．具体的には，x，y，z_i，z'_a の四つの空間で検索をし，比較を行った．ここでは属性としては眼鏡に着目し，CFGAN としては 3 次元の SB-I を用いた．実験結果を図 9 に示す．結果からは，x や z_i は属性が考慮されていない空間であるため，検索画像に眼鏡が含まれているとは限らないが，y や z'_a は属性を考慮した空間であるため，眼鏡を含む画像を見つけ出すことができていることが分かる．さらに，y は属性の表現力が低い（一次元表現）ため眼鏡のタイプまで一致した画像を見つけ出すことはできないが，z'_a は属性に対して高い表現力（多次元表現）を持つため，眼鏡のタイプ（例，細ぶち眼鏡，サングラスなど）まで一致した画像を見つけ出すことが可能であることが分かる．なお，この属性表現は教師データとして与えたものではなく，CFGAN の学習の過程で自動的に獲得したものであることに留意されたい．

7. おわりに

本稿では，生成的属性制御と呼ぶ新しい問題を提起し，この問題を解くために CFGAN と呼ぶ CGAN の新しい拡張モデルを提案した．さらに，本稿では典型的な GUI を用いて制御するためのフィルタリング構造と隠れ変数の分布形状を定義し，表現力と操作性のある属性の表現空間を得ることを可能にした．実験では，属性に基づく画像生成，編集，転写，検索の四つのタスクに CFGAN を適用し，有効性を示した．CFGAN の限界としては，z'_a の各次元は教師なしで学習されるため，事前に名前をつけることができない点が挙げられるが，実験で示したように各次元の意味は一貫性があるため，後から名前をつけることは可能である．本研究の拡張方法としては，他の GUI コントローラの実現，より高次元のデータへの適用などが考えられる．

参考文献

[1] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y.: Generative adversarial nets, *NIPS* (2014).
[2] Kingma, D. P. and Welling, M.: Auto-encoding variational Bayes, *ICLR* (2014).
[3] Rezende, D. J., Mohamed, S. and Wierstra, D.: Stochastic backpropagation and approximate inference in deep generative models, *ICML* (2014).
[4] Mirza, M. and Osindero, S.: Conditional generative adversarial nets, *arXiv preprint arXiv:1411.1784* (2014).
[5] Guo, D. and Sim, T.: Digital face makeup by example, *CVPR* (2009).
[6] Liu, L., Xu, H., Xing, J., Liu, S., Zhou, X. and Yan, S.: Wow! You are so beautiful today!, *ACMMM* (2013).
[7] Tong, W.-S., Tang, C.-K., Brown, M. S. and Xu, Y.-Q.: Example-based cosmetic transfer, *PG* (2007).
[8] Yang, F., Wang, J., Shechtman, E., Bourdev, L. and Metaxas, D.: Expression flow for 3D-aware face component transfer, *SIGGRAPH* (2011).
[9] Hassner, T., Harel, S., Paz, E. and Enbar, R.: Effective face frontalization in unconstrained images, *CVPR* (2015).
[10] Kemelmacher-Shlizerman, I., Suwajanakorn, S. and Seitz, S. M.: Illumination-aware age progression, *CVPR* (2014).
[11] van den Oord, A., Kalchbrenner, N. and Kavukcuoglu, K.: Pixel recurrent neural networks, *ICML* (2016).
[12] Kingma, D. P., Mohamed, S., Rezende, D. J. and Welling, M.: Semi-supervised learning with deep generative models, *NIPS* (2014).
[13] Yan, X., Yang, J., Sohn, K. and Lee, H.: Attribute2Image: Conditional image generation from visual attributes, *ECCV* (2016).
[14] Odena, A., Olah, C. and Shlens, J.: Conditional image synthesis with auxiliary classifier GANs, *ICML* (2017).
[15] van den Oord, A., Kalchbrenner, N., Espeholt, L., Vinyals, O., Graves, A. and K, K.: Conditional image generation with PixelCNN decoders, *NIPS* (2016).
[16] Kulkarni, T. D., Whitney, W. F., Kohli, P. and Tenenbaum, J.: Deep convolutional inverse graphics network, *NIPS* (2015).
[17] Makhzani, A., Shlens, J., Jaitly, N. and Goodfellow, I.: Adversarial autoencoders, *NIPS* (2016).
[18] Radford, A., Metz, L. and Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks, *ICLR* (2016).
[19] Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I. and Abbeel, P.: InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets, *NIPS* (2016).
[20] Farhadi, A., Endres, I., Hoiem, D. and Forsyth, D.: Describing objects by their attributes, *CVPR* (2009).
[21] Kumar, N., Berg, A. C., Belhumeur, P. N. and Nayar, S. K.: Attribute and simile classifiers for face verification, *ICCV* (2009).
[22] Lampert, C. H., Nickisch, H. and Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer, *CVPR* (2009).
[23] Parikh, D. and Grauman, K.: Relative attributes, *ICCV* (2011).
[24] Yu, A. and Grauman, K.: Just noticeable differences in visual attributes, *ICCV* (2015).
[25] Zhu, J.-Y., Krähenbühl, P., Shechtman, E. and Efros, A. A.: Generative visual manipulation on the natural image manifold, *ECCV* (2016).
[26] Dosovitskiy, A. and Brox, T.: Generating images with perceptual similarity metrics based on deep networks, *NIPS* (2016).
[27] Johnson, J., Alahi, A. and Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution, *ECCV* (2016).
[28] Brock, A., Lim, T., Ritchie, J. and Weston, N.: Neural photo editing with introspective adversarial networks, *ICLR* (2017).
[29] LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P.: Gradient-based learning applied to document recognition, *Proc. of the IEEE*, Vol. 86, No. 11, pp. 2278–2324 (1998).
[30] Wah, C., Branson, S., Welinder, P., Perona, P. and Belongie, S.: The Caltech-UCSD birds-200-2011 dataset, Technical report (2011).
[31] Liu, Z., Luo, P., Wang, X. and Tang, X.: Deep learning face attributes in the wild, *ICCV* (2015).
[32] Eigen, D., Puhrsch, C. and Fergus, R.: Depth map prediction from a single image using a multi-scale deep network, *NIPS* (2014).
[33] Krizhevsky, A., Sutskever, I. and Hinton, G. E.: ImageNet classification with deep convolutional neural networks, *NIPS* (2012).
[34] Nair, V. and Hinton, G. E.: Rectified linear units improve restricted Boltzmann machines, *ICML* (2010).
[35] Maas, A., Hannun, A. Y. and Ng, A. Y.: Rectifier nonlinearities improve neural network acoustic models, *ICML* (2013).
[36] Xu, B., Wang, N., Chen, T. and Li, M.: Empirical evaluation of rectified activations in convolutional network, *ICML Workshop* (2015).
[37] Ioffe, S. and Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift, *ICML* (2015).
[38] Kingma, D. P. and Welling, M.: Adam: A method for stochastic optimization, *ICLR* (2015).
[39] Larsen, A. B. L., Sønderby, S. K. and Winther, O.: Autoencoding beyond pixels using a learned similarity metric, *ICML* (2016).