CONFIDENTIAL EXTENDED ABSTRACT.
DO NOT DISTRIBUTE ANYWHERE.

第 21 回 画像の認識・理解シンポジウム

単語情報を利用した画像の質感転送

杉山 優[1,a)]　柳井 啓司[1,b)]

概要

現在、画像変換の主な手法としてスタイル変換がある。しかしこれはスタイル画像とコンテンツ画像を時間をかけて最適化しなければ変換が行えない。改良手法の高速スタイル変換では予め学習したスタイル画像が決まっていればどのような画像も似たような変換になってしまう。そこで、本研究では言葉による画像変換を実装すると同時に、コンテンツ画像の物体認識を行った。画像を言葉によって任意に変換することが可能となった。

1. はじめに

画像変換の技術として代表的なものにスタイル変換 [1] と呼ばれる技術がある。しかしこれはスタイル画像とコンテンツ画像の二枚をネットワークに入力して時間をかけて最適化しなければ変換が行えず、手軽な手法ではない。またスタイル画像が決まっていればどのような画像も似たような変換になってしまうため、コンテンツ画像に似合わないスタイル変換を行ってしまうことも多い。

スタイル変換の改良手法は数多く存在していて、変換を高速化するもの [3] や、高精度に変換を行うもの [4] などが提案された。しかし、これらの多くは変換の前にスタイル画像単独での学習を必要とし、汎用的にスタイル画像を使用できない。そこでこの解決策として提案された手法に任意スタイル変換 [7]、[2] がある。この手法では事前にスタイルを学習して変換自体は高速に行うという方針は変えずに、学習したスタイルをベクトル表記していくことによってあらゆる画像をスタイル画像のベクトルとして変換に使用することができるようになるというものである。

しかし、スタイル画像が適切ではなかったときに、変換がうまく行かなくなってしまうという欠点を抱えている。そこで入力された画像によって変換を変化させ、画像に適切な画像変換を行うような手法が必要となる。

2. 関連研究

画像変換の手法にはいくつかあるが、その中でもコンテンツ画像に対してのスタイル変換を行っているものとして下の研究がある。

2.1 スタイル変換

スタイル変換の手法として代表的で最もプリミティブなものに Gatys らの Imagestyle transfer using convolutional neural networks [1] がある。この手法では、コンテンツ画像とスタイル画像の二種類を用意し、２つの画像どちらにも近づくような画像を作成するようにニューラルネットワークを学習させることで、コンテンツ画像に映っているものは維持しつつ、スタイル画像のような画像の風味に変換する。

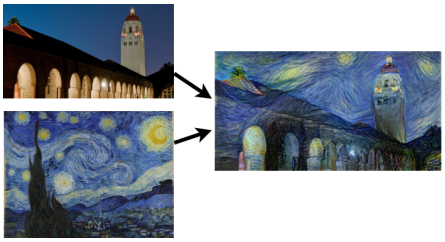

図 1　スタイル変換の様子

変換の学習は、スタイル画像とコンテンツ画像それぞれのロス関数を足し合わせたものを全体のロスとして学習を行う。また、変換には VGG ネットワーク [6] を使用する。VGG の FC 層を除いた部分のパラメータを使用する。画像識別ネットワークの深い層では画像の意味を持った部分だけが判別され、物体認識にはあまり影響しない模様等をスタイル画像に近づけることでスタイル変換を行うことができる。

コンテンツ画像のロス関数は下のようになっている。$\vec{p}$、$\vec{x}$ はそれぞれオリジナルの入力されたコンテンツ画像と生成画像を、P^l、F^l はそれぞれの画像のレイヤ l における特徴ベクトルを示す。

$$L_{content}(\vec{p},\vec{x},l) = \frac{1}{2}\sum_{i,j}(F^l_{ij} - P^l_{ij})^2 \qquad (1)$$

スタイル画像のロス関数は下のように計算する。まずはグラム行列 G^l、A^l をスタイル画像と生成画像でそれぞれ計算し、スタイル画像のロスを計算する。ある層のみについ

1　電気通信大学大学院 情報理工学研究科 情報学専攻
a)　sugiya-y@mm.inf.uec.ac.jp
b)　yanai@cs.uec.ac.jp

て計算するのではなく、複数の層を加算して考える。

$$G^l_{ij} = \sum_k F^l_{ik} F^l_{jk} \tag{2}$$

$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} (G^l_{ij} - A^l_{ij}) \tag{3}$$

$$L_{style}(\vec{a}, \vec{x}) = \sum_{l=0}^{L} w_l E_l \tag{4}$$

これらを足し合わせ、学習で最適化していくロス関数は下のようになる。

$$L_{total}(\vec{p}, \vec{a}, \vec{x}) = \alpha L_{content}(\vec{p}, \vec{x}) + \beta L_{style}(\vec{a}, \vec{x}) \tag{5}$$

つまりスタイル画像とコンテンツ画像が互いに近づくように最適化していくため、画像の組が変わるたびに時間をかけて画像を最適化する必要がある。これでは画像を入力するたびに長い時間をかけることになるために、変換画像を直ぐに用意したいといった要求に答えることができない。それに対応する研究に高速スタイル変換というジャンルがある。

2.2 高速スタイル変換

スタイル変換を高速化する手法として、Fast Neural Style Transfer [3] がある。事前にスタイルを学習しておくことで、新たなコンテンツ画像が入力された際に、学習済みのスタイルに変換するというネットワークを構築するものである。

しかし、高速スタイル変換では事前に学習したスタイル画像しか変換に使用できず、新たなスタイルで変換がしたいという要求にはその都度長い時間をかけてスタイル画像を学習しなければならない。

2.3 任意スタイル変換

スタイル変換の改良手法が多く提案されている中、学習の汎用性を持たせる研究として Yanai の Unseen Style Transfer [7] や Exploring the structure of a real-time, arbitrary neural artistic stylization network [2] がある。従来の Jonhson らの高速スタイル変換 [3] では、学習に用いた 1 種類のスタイル画像のみのスタイル変換しかできなかったが、この研究では、事前に学習に使っていないスタイルも含めて任意のスタイルの変換が可能となるネットワークを提案している。ネットワークは図 2 のようになっている。

高速スタイル変換 [3] と同様に、変換を行う前にスタイルだけを学習しておき、実際にコンテンツ画像を入力して変換する際には高速に動作させることができる。さらに高速スタイル変換 [3] とは異なり、学習の際に複数画像を用いてスタイル画像から Style Vector を生成する Style Condition Network を学習することにより、任意のコンテンツ画像に対して変換を行うことができる。

事前に複数の画像に対してのスタイルを学習して、任意のスタイルに対応できるネットワークを作成しているために、変換の際にはスタイル画像とコンテンツ画像の両者はどんな画像でも対応でき、高速な変換が可能であることになる。

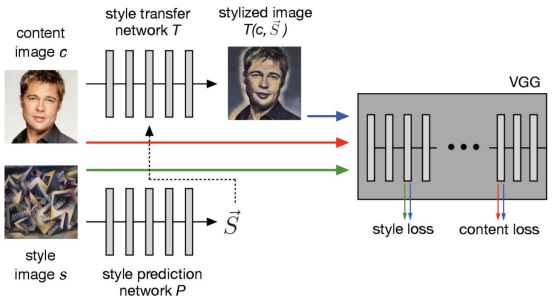

図 2 任意スタイル変換のネットワーク図 ([2] から引用)

3. 手法概要

本研究では、Unseen Style Transfer のネットワークの一部を入れ替えるように学習を行うことで画像の変換を実装した。以下に本研究での画像変換の学習と変換の流れを示す。

(1) スタイル画像とその特徴を示す単語の組になったデータセットを用意

(2) Word2Vec [5] と VGG [6] による Style Selector Network(図 3) を用意

(3) Style Selector Network を Unseen Style Transfer の中間ネットワークの Style Conditon Network(図 [7]) と近似するように学習

(4) 学習した Style Selector Network(図 3) を Arbitrary Style Transfer の中間ネットワークとして Style Conditon Network(図 2) の代わりに入力し、画像を生成

4. 手法詳細

4.1 データセットの構築

データセットとして UFMD[*1]の質感画像を使用した。画像は対応する単語にそれぞれ 10000 枚存在する。ラベルは「布」、「植物」、「ガラス」、「革」、「金属」、「紙」、「プラスチック」、「石」、「水」、「木」の全１０種類である。データセットに使用した画像の例を図 4 に示す。

4.2 Word2Vec の学習

文字情報を変換のパラメータとして使用するためには、入力された文字列をベクトルで表現しなければならない。それを行うために Word2Vec [5] を使用する。

[*1] https://people.csail.mit.edu/celiu/CVPR2010/FMD/

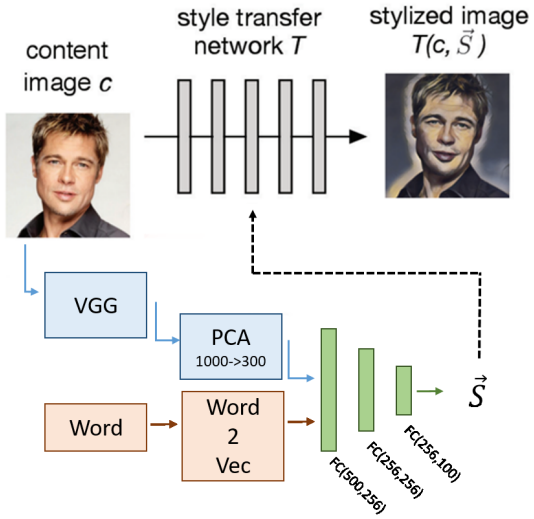

図 3　Style Selector Network

図 4　質感画像データセットに使用した画像例

日本語の Word2Vec モデルは公開されていないため、モデルを学習する必要がある。本研究では、容易に入手可能である日本語 WikiPedia データ (3.1GB) を利用して，日本語 Word2Vec モデルの学習を行った。

4.3　VGG による画像認識

VGG16 [6] を使用して、画像特徴量を取得した。今回は画像の意味特徴を表現していることを期待してこのネットワークの fc8 層にある 1000 次元の全結合レイヤを使用した。

取り出した中間層のベクトルはそのままでは 1000 次元のベクトルだが、これでは Word2Vec [5] の 200 次元というベクトルの次元数と釣り合いが取れず、実際に学習した際に画像特徴を重視しすぎてしまうという良くない傾向が出たために、VGG からのベクトルを次元圧縮した。次元圧縮には PCA を使用し、300 次元までベクトルの次元を削減した。

4.4　Style Selector Network の構成

VGG 画像認識によって得た画像からの中間信号と単語を Word2Vec に入力して得たワードパラメータを結合し 500 次元のベクトルに整形し、それを図に示す FC 層が 3 層で各層に活性化関数 Relu を使用し、Dropout を挟んだ Style Selector Network(図 3) に通した出力が Style Prediction Network(図 2) の出力に近づくように収集した画像を用いて学習した。

学習したモデルを使用して画像を変換する際には、Arbitrary Style Transfer のネットワークのうち、Style Prediction Network 部分を Style Selector Network(図 3) に差し替えたネットワークを使用して、スタイル画像を入力する代わりに、変換したい言葉を入力することによって変換を行った。

学習に使用した Style Selector Network のロス関数 L_{SNN} は下のようになっている。L2 距離を最小化することで学習を行った。v_i と w_i はそれぞれ画像特徴と単語特徴のベクトルを示し、s_i は Style Vector を示す。

$$L_{SNN} = \frac{1}{2}\sum_i (s_i - SNN(v_i, w_i))^2 \tag{6}$$

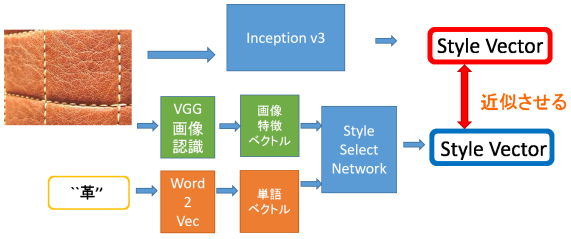

図 5　学習の概略

5.　実験

実際に画像を学習したネットワークに通して変換を実験した。実験は下に示す 2 通り行った。各実験について、上手く変換できた例とできなかった例を示す。

(1) 学習に使用した単語を入力して変換した場合
(2) 学習に使用しなかった単語を入力して変換した場合

5.1　学習に使用した単語を入力して変換した場合

図 6 に学習に使用した単語を利用して画像を変換することで、このネットワークの画像変換能力を観察した様子を示す。全体的に色数が増え、入力単語が変わることによって色やテクスチャ等のスタイルが転送されていることがわかった。木箱や海の画像の質感転送は特にうまく画像の変化が現れていて、明らかに質感が変化していると言えるほどの変化が起こっている上に、単語間での変化の様子も異なる。

5.2　学習に使用しなかった単語を入力して変換した場合

次に、学習に使用しなかった単語を入力して変換を行った場合に、学習に使用した単語とどのように異なるかを観察するために実験を行った。図 7 に示す通り、学習に使用しなかった単語でも似た意味の学習済み単語と似たような変換が行われた。しかし単語が変わることで少しのスタイ

図 6　学習に使用した単語で画像を変換した様子

ルの変化が生じることも観察できた。“アクリル” の変換では “ガラス” や “プラスチック” と似たような変換になり、“レンガ” の変換では “石” と似たような変換になっていることがわかる。

図 7　質感画像スタイルで画像を変換した様子

どちらの実験においても、いちごの画像のようにスタイルの変化が少ないコンテンツ画像というものが存在し、シンプルに物体があるだけの木箱のような画像では変換が単語ごとに大きく異なる。また、建物の画像においてはスタイルの転送は色のみにとどまり、画像全体にブラーがかかってしまっていて、精細な画像の生成が行えていない。

6. 考察

うまく変換できた例とできなかった例を見比べると、建物のような画像では上手く変換できていない。これは VGG 画像認識によって画像の特徴を認識したからだと考えられる。VGG 画像認識は画像に写っている物体についての認識をするものであり、風景などの “シーン” を判別することは難しい。これによって “猫” や “木箱” などの物体は認識できても “海原” や “ビル” などの背景になるような画像は認識できないという現象が起こる。これを解決するためには VGG 画像認識によって作成しているベクトルを、シーン認識のネットワークを使用して生成することによって解決することが出来ると考えられる。

また、“ガラス” の変換などは色が多くつくだけ、という変換になっているものが多いことが観察できた。これは、データセットとして UFMD を使用し、このデータセットの中でガラスカテゴリにはかなり様々な色や形状のガラスが含まれているために、うまく共通とくちょくを学習することができなかったと考えられる。これを解決するためには更にデータセットを拡張する必要があると考えている。

7. Conclusion

本提案手法によって、“言葉による画像変換” と “コンテンツ画像に適した画像変換” の 2 つの目的は達成できた。特に VGG 画像認識で上手く認識できるような、画像の中心にコンテンツが表示されている画像に関しては良い結果が得られて、逆に背景画像のような物体認識の行えない画像に関しては上手く行かなかった。

今後はシーン認識によって背景を示す画像に対応すること、各単語の画像を増やすことや、学習に使用する単語の種類を増やすことでより大きなデータセットの構築を行うことや、ユーザー評価を行うことで客観的な結果の考察などを行っていきたい。

謝辞: 本研究は JSPS 科研費 17H01745/15H05915/17H05972/17H06026/17H06100 の助成を受けたものです。

参考文献

[1] Gatys, A., Ecker, S. and Bethge., M.: Image style transfer using convolutional neural networks, *Proc.of IEEE conference on computer vision and pattern recognition*, pp. 2414–2423 (2016).

[2] Ghiasi, G., Lee, H., Kudlur, M., Dumoulin, V. and Shlens, J.: Exploring the structure of a real-time, arbitrary neural artistic stylization network, *Proc. of British Machine Vision Conference* (2017).

[3] Johnson, J., Alahi, A. and Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution, *Proc.of European Conference on Computer Vision*, pp. 694–711 (2016).

[4] Liao, J., Yao, Y., Yuan, L., Hua, G., Bing, A. S. and , K.: Visual Attribute Transfer through Deep Image Analogy, Vol. arXiv:1705.01088 (2017).

[5] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. and Dean., J.: Distributed representations of words and phrases and their compositionality, *Proc.of Advances in Neural Information Processing Systems 25*, pp. 3111–3119 (2013).

[6] Simonyan, K., Vedaldi, A. and Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition, *Proc.of International Conference on Learning Representation* (2015).

[7] Yanai, K.: Unseen Style Transfer Based on a Conditional Fast Style Transfer Network, *Proc.of International Conference on Learning Representation Workshop Track (ICLR WS)* (2017).