

CNN を用いた画風変換による物体検出精度の改善

高橋沙季[†] 竹内健[†] 金井謙治[†] 甲藤二郎[†]

概要: 本研究では、画風変換による不鮮明画像の鮮明化を扱う。一般的に、霧や霏などの HAZE 発生時に撮影された画像を見やすくするためには、ダークチャネルを用いた鮮明化手法が主流である。しかし、この手法では HAZE が強い場合には鮮明化効果が発揮されない。そこで本研究では、HAZE が強い場合にも鮮明化を実現させるため、CNN を用いた画風変換により不鮮明画像を鮮明化する。具体的には、HAZE 発生時に撮影された不鮮明画像を、晴れの日の昼間に撮影された別の画像の画風に似せて画風変換することにより、鮮明化を行う。また、本研究では鮮明化効果に加えて、画像に写る物体の検出精度向上も図る。

キーワード: 画風変換, HAZE 除去, 物体検出, Deep Learning

Improvement of Object Detection Accuracy by Style Transformation using Convolutional Neural Network

SAKI TAKAHASHI[†] MASARU TAKEUCHI[†]
KENJI KANAI[†] JIRO KATTO[†]

Abstract: In this paper, we improve sharpness of hazy images by style transformation. Generally, the use of dark channel is the most well-known method to alleviate hazy noise of images. However, in this method, the sharpening effect is limited as the hazy noise becomes strong. Therefore, in this research, in order to improve the sharpness of strong hazy images, we apply a state-of-the-art style transformation approach using Convolutional Neural Network (CNN). Specifically, we sharpen the hazy images by applying style transform of daytime images which are clearly captured. In addition to the sharpening effect, our research also aims to improve the detection accuracy of objects appearing in images.

Keywords: Style transformation, Hazy-noise reduction, Object detection, Deep Learning

1. はじめに

近年、機械学習や Deep Learning の研究開発が盛んに行われている。Deep Learning は音声認識や囲碁プログラムなど様々な分野に応用されており、我々の Quality of Life (QoL)を高める上で今後深く浸透すると予想され、また娯楽を支えるものとしても大きく貢献していくと期待される。後者の一例に画風変換という技術がある。画風変換は、ある画像に写る物体の配置をそのままに、その画像の画風を全く別の画像の画風に似せて変換する技術である。

本研究ではこの画風変換を、生成される画像の美しさを楽しむためではなく、不鮮明画像の鮮明化を目的として使用する。具体的には、霧や霏などの HAZE 発生時に撮影された画像を、晴れの日の昼間に撮影した別の画像の画風に似せて変換することにより、画像の視認性向上を図る。さらに、一般物体認識技術にも適用することで、不鮮明画像に対する物体の検出精度改善も図る。

2. 一般的な HAZE 除去手法

“HAZE”とは視界が悪くなる現象全般を指す気象用語で

あり、特に霧や霏を指す場合が多い。HAZE 除去手法にはコントラストを最大化する手法[1]や、物体の拡散光を推定する手法[2]など様々なものがあるが、広く知られている手法の一つとして He らによる HAZE 画像モデルとダークチャネルを用いた手法[3]が挙げられる。

HAZE 画像モデルでは、入力 HAZE 画像から環境光と透過マップを推定することで、HAZE を除去した鮮明画像を復元することができる。しかし一つの既知数から三つの未知数を決定する必要があるため、計算解を一意に得ることができない。そこで He らは、HAZE 画像のダークチャネル強度が HAZE 濃度を示すことを利用し、HAZE 画像の空領域を探索することによって環境光および透過マップを推定する手法を提案している。

以下に、HAZE 画像モデル、ダークチャネル、He らの手法の流れについて述べる。

2.1 HAZE 画像モデル

HAZE 画像モデルは、以下の式(1)で表される[4].

$$I(x) = J(x)t(x) + A(1 - t(x)) \quad \text{式(1)}$$

x :画素位置 $I(x)$:HAZE 画像

[†] 早稲田大学
Waseda University

$J(x)$:鮮明画像 (HAZE 除去画像)
 $t(x)$:透過マップ A :環境光

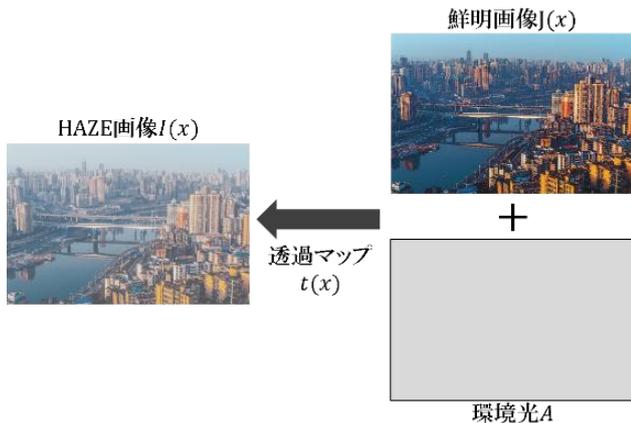


図1 HAZE 画像モデル[4]

また, HAZE 画像モデルを図1に示す. $t(x)$ は以下の式(2)で表される透過マップで, 情景放射による光が霧や霏の粒子に散乱されずにカメラに到達する度合いを示す. また A は環境光で, 全ての画素で同一である. このモデルでは, $J(x)t(x)$ を直接減衰と呼び, 情景情報が散乱なくカメラに到達する度合いを示す. また $A(1-t(x))$ は環境光が HAZE 画像に及ぼす影響の度合いを示す.

$$t(x) = e^{-\beta d(x)} \quad \text{式(2)}$$

β :環境光の散乱係数

$d(x)$:物体からカメラまでの距離

環境光による減衰は, 光が通過する距離が長くなるほど大きくなるため, 透過マップは近景の物体が写っているときは値が大きく, 遠景の物体では値が小さくなる. HAZE 画像においては, 透過マップは HAZE 濃度を表し, HAZE 濃度は遠景で高く, 近景で低くなる.

2.2 ダークチャンネル

ダークチャンネルは, 以下の式(3)で表される.

$$I^{dark}(x) = \min_{c \in \{r, g, b\}} (\min_{y \in \Omega(x)} I^c(y)) \quad \text{式(3)}$$

I^c : I の色チャンネル

$\Omega(x)$: x を中心とする局所領域

Heらによると, 5000枚の日中の HAZE 無し画像を用いた実験結果から, 全画素の約75%はダークチャンネル強度が0であり, 90%のダークチャンネル強度が25に満たないことが分かっている. 一方 HAZE 画像の場合は環境光が加わるため, 鮮明画像よりも明るくなり, ダークチャンネル強度は高くなる. したがってダークチャンネル強度は HAZE 濃度を示し, 物体までのおおよその距離も示すことになる.

2.3 Heらによる HAZE 除去手法の流れ

Heらによる HAZE 除去手法は, 2.1の HAZE 画像モデルと 2.2のダークチャンネルに基づくものである. Heらは, HAZE 画像のダークチャンネル強度が HAZE 濃度を示すことを利用し, HAZE 画像の空領域を特定することによって環境光を推定した上で, 透過マップ推定および鮮明画像復元を行う手法を提案している. 環境光の推定, 透過マップ推定および鮮明画像復元は次のように行われる.

2.3.1 環境光の推定

2.1の HAZE 画像モデルからダークチャンネルを得る場合, 式(4)のように表現される.

$$I^{dark}(x) = J^{dark}(x)t(x) + A^c(1-t(x)) \quad \text{式(4)}$$

x :画素位置

$I^{dark}(x)$: HAZE画像のダークチャンネル

$J^{dark}(x)$:鮮明画像のダークチャンネル

A^c :環境光の色チャンネル

鮮明画像のダークチャンネルのほとんどの輝度値は0であるため, 式(4)は以下の式(5)のように書き換えることができる. また式(5)より, 透過マップを推定する式(6)を得ることができる.

$$I^{dark}(x) \approx A^c(1-t(x)) \quad \text{式(5)}$$

$$t(x) \approx 1 - I^{dark}(x)/A^c \quad \text{式(6)}$$

ここで, HAZE 画像の空領域については距離が無限遠であるため, この領域の $t(x)$ はほぼ0になり, 式(7)に示す通り環境光 A の値は空の色と等しいとすることができる.

$$A^c = I^{dark}(x) \quad \text{式(7)}$$

このように, 空領域を特定することにより環境光を推定する. 具体的には, I^{dark} の上位0.1%の高強度画素を抽出し, この画素のうち HAZE 画像 $I(x)$ の輝度値が最も高い画素を環境光として選び出す.

2.3.2 透過マップ推定

2.3.1のように環境光を推定した後は, 透過マップを推定する. 入力 HAZE 画像から得られる粗マップ及び詳細マップを合成することにより, 各画素の HAZE 濃度を表す透過マップを得ることができる. 以降, 詳細に説明する.

粗マップとは局所領域に基づくダークチャンネルのことであり, 以下の式(8)で表される.

$$M^{coarse}(x) = \min_{c \in \{r, g, b\}} (\min_{y \in \Omega(x)} I^c(y)) \quad \text{式(8)}$$

一方, HAZE 画像のエッジ詳細情報を表す詳細マップは次式(9)で表される.

$$M^{fine}(x) = \min_{c \in \{r, g, b\}} I^c(x) \quad \text{式(9)}$$

透過マップを得るためには, これらの粗マップと詳細マップを合成する必要がある. 合成マップは式(10)に示すと

おり、詳細マップのブロックの最大値を探し、詳細マップの位置 x にある画素値と比較して小さいほうを選び出すことによって算出される。また、合成マップから透過マップを算出することができ、式(11)で表される。なお、 ω は画像の HAZE 除去の度合いを表すパラメータであり、0 より大きく 1 より小さい。この値は 0.9 が適切とされている。

$$M^t(x) = \text{式(10)}$$

$$\min(\max_{y \in \Omega(x)} M^{\text{coarse}}(y), M^{\text{fine}}(x))$$

$$t(x) = 1 - \omega M^t(x)/A \quad \text{式(11)}$$

2.3.3 鮮明画像の復元

2.3.1 のように環境光を推定し、2.3.2 のように透過マップを推定することにより、次式(12)に従って鮮明画像を復元することができる。

$$J(x) = \frac{I(x) - A}{\max(t(x), t_0)} + A \quad \text{式(12)}$$

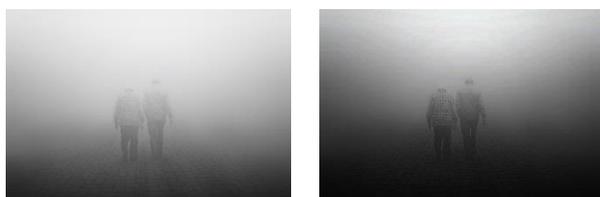
$t_0: 0$ による除算を避けるための定数値

図2に HAZE 画像 I , HAZE 画像のダークチャンネル I^{dark} , HAZE 画像の透過マップ t , 鮮明画像 J の一例を示す。



図2 ダークチャンネルを用いた鮮明化手法の実験画像例

しかし He によるダークチャンネルを用いた鮮明化手法では、環境光を空領域の色と仮定するため、HAZE が強い場合には図3のように鮮明化効果を十分に得られないという課題がある。



(a) 鮮明化前 (b) 鮮明化後

図3 HAZE が強い場合における He らの手法による不鮮明画像の鮮明化結果

3. 画風変換

画風変換は 2015 年 9 月に Gatys らによって発表された研究分野[5]であり、図4に示すように、Content Image に写る物体の配置をそのままに、その画像の画風を全く別の画像 Style Image の画風に似せて変換する技術である。アルゴリズムには Deep Learning の一種である Convolutional Neural Network (CNN) を使用しており、誤差逆伝播法を用いて画像を少しずつ修正し、変換後画像を生成する。



図4 画風変換画像[6]

現在最も主流な画風変換手法の一つは、2016 年 3 月に Johnson らが発表した論文[7]で紹介されているモデルを用いた手法である。Johnson らの手法では、図5のように VGG-16CNN モデルの損失計算ネットワークと、16 層 CNN モデルの画風変換ネットワークを用意し、損失計算ネットワークで中間層における損失を計算して、その損失を最小化するように画風変換ネットワークの重みを更新する[8]。一枚の Style Image に対して数万枚の Content Image を用いて画風変換ネットワークを学習することにより、任意の画像の高速な画風変換が可能となる。なお、Style Image 一枚ごとに画風変換ネットワークを新しく学習する必要がある。

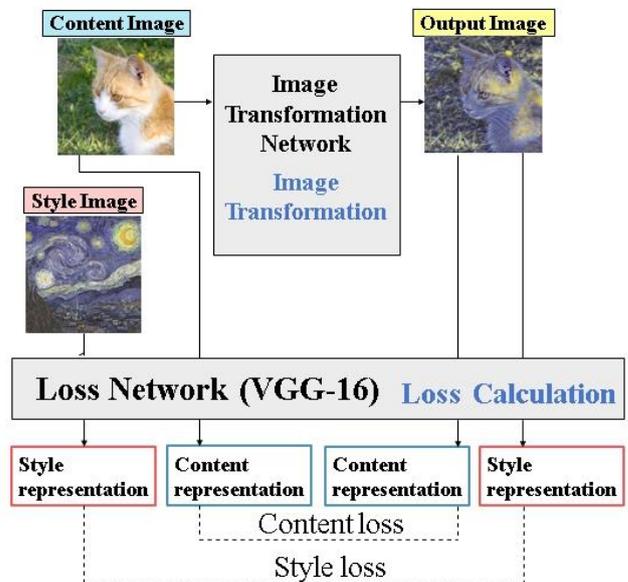


図5 画風変換モデル[9]

ここで損失は、二枚の画像のずれ具合と言い換えることができ、図 5 中の Content Loss (中間層における Content Image と Output Image の損失) および Style Loss (中間層における Style Image と Output Image の損失) は以下の式(13)~(15)で表される. この Content Loss と Style Loss の和を最小化する方向に画風変換ネットワークの重みを更新することにより, Content Image にも Style Image にも似た画像を生成することが可能となる.

$$L_{content}^{\phi,j}(\hat{y}, y_c) = \frac{1}{C_j H_j W_j} \|\phi_j(\hat{y}) - \phi_j(y_c)\|^2 \quad \text{式(13)}$$

$$G_j^{\phi}(x)_{c,d} = \frac{1}{C_j H_j W_j} \sum_{h=1}^{H_j} \sum_{w=1}^{W_j} \phi_j(x)_{h,w,c} \phi_j(x)_{h,w,c'} \quad \text{式(14)}$$

$$L_{style}^{\phi,j}(\hat{y}, y_s) = \|G_j^{\phi}(\hat{y}) - G_j^{\phi}(y_s)\|^2 \quad \text{式(15)}$$

\hat{y} : y を画風変換した画像

y_c : コンテンツ画像

y_s : スタイル画像

c : あるチャンネル

c' : 別のチャンネル

$C_j H_j W_j$: レイヤ j のチャンネル数, 高さ, 幅

ϕ_j : x を Loss Network に入力した時のレイヤ j の値

4. 画風変換を用いた HAZE 除去手法

本稿では, 画風変換を用いた HAZE 除去手法を提案する. 前章の画風変換モデルにおいて, 図 6(a)に示すような HAZE 画像を Content Image とし, Style Image に晴れの日の昼間に撮影した画像を入力することにより, Content Image の鮮明化を図る.

事前準備として, 一枚の Style Image と数万枚の Content Image を用いて, 未知の画像を入力すると高速に画風変換を行う画風変換ネットワークの学習を行う. 学習にあたって, Content Image には Microsoft COCO データセット[10]に含まれるおよそ八万枚の画像を使用し, Style Image には図 6(b)の画像を用いる. Microsoft COCO データセットに含まれる画像の一例を図 7 に示す.



(a) Content Image

(b) Style Image

図 6 HAZE 除去の Content Image と Style Image 例

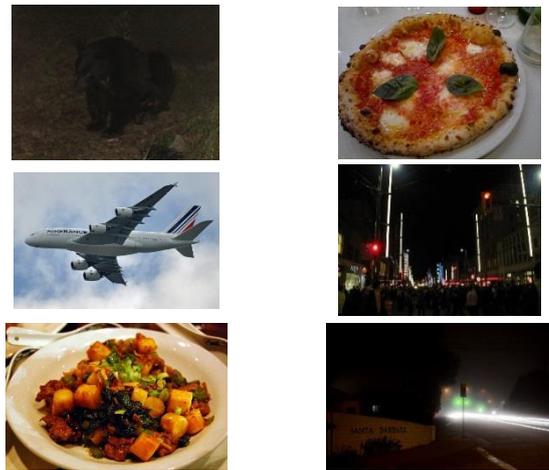
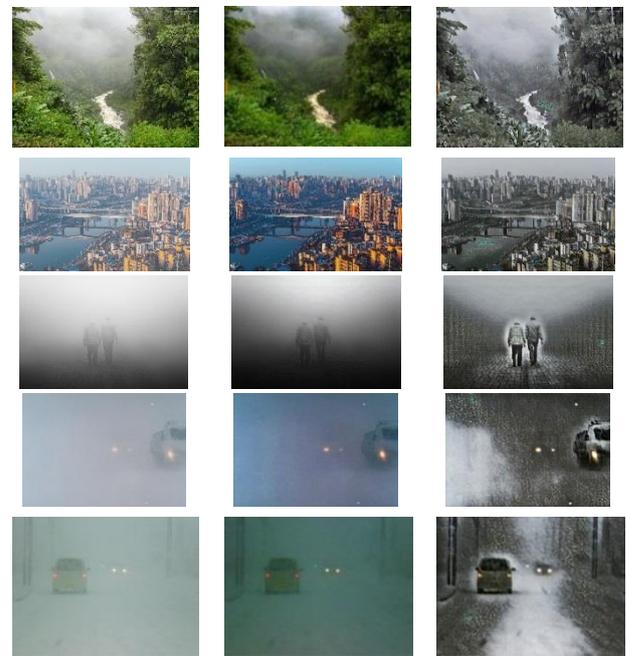


図 7 Microsoft COCO dataset に含まれる画像例 [10]

5. 比較実験の結果と考察

He らの HAZE 除去手法と画風変換による HAZE 除去手法の比較実験を行った. 実験結果例を以下の図 8 に示す. 図 8 は左から順に入力 HAZE 画像[11][12], He らの手法による鮮明化結果, 画風変換による鮮明化結果である.



入力 HAZE 画像

ダークチャンネル

画風変換手法による結果

手法による結果

よる結果

図 8 He らの手法と画風変換手法の比較実験結果例

He が提案するダークチャンネルを用いた手法では, 図 8 に示す 3 から 5 番目の画像のように, 鮮明化の効果が低くなってしまふ場合がある. この原因として, 図 9 のダークチャンネルおよび透過マップ画像を見てみると, これらの 3 画像 (図 9 に示す 3 から 5 番目の画像) は全体的にダークチャンネル強度が高く, 透過マップの値が小さいことから, HAZE 濃度が高い画像と言える.



ダークチャネル 透過マップ
図9 ダークチャネル画像と透過マップ画像

Heらの手法では、環境光は空領域の色に等しいという仮定を立てているため、HAZEが強い場合には空領域が正確に特定されず、環境光の値を正しく推定することができない。この理由から、空領域がよく見えない画像（濃霧画像など）にはHeらの手法は不向きであると考えられる。

一方画風変換手法は図8のとおり、HAZEが強い場合にも鮮明化効果が得られている。変換後に原画像の色味を保つことができないというデメリットはあるが、人物や物体の検出問題として扱う分には問題がないと考えられる。実際に濃霧画像において、Heらの手法よりも画風変換手法による鮮明化のほうが、人物や車両の検出精度が高くなる例を確認している。図10にその一例を示す。なお、人物検出にはYOLOv2[13][14]を用いている。

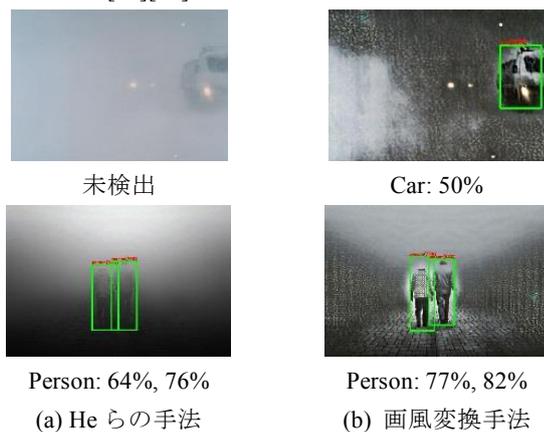
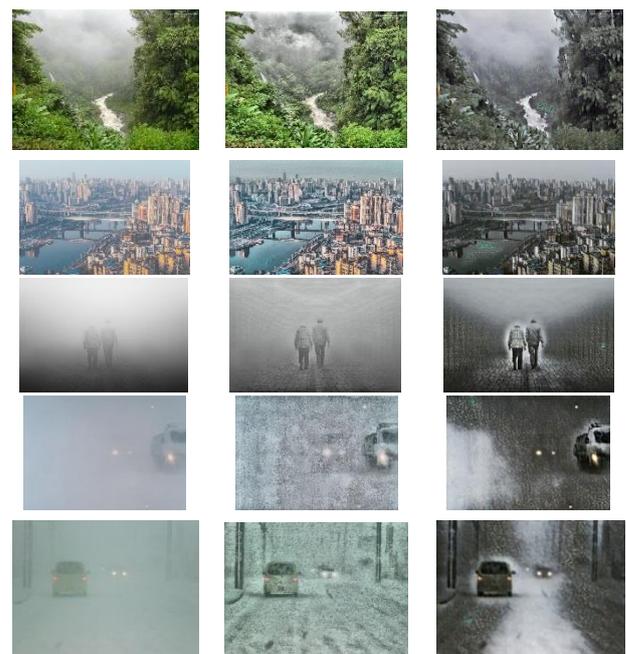


図10 物体検出結果

6. 鮮明化機器による鮮明化結果との比較実験

最後に、市販の鮮明化機器による鮮明化と、画風変換手法による鮮明化の比較実験を行う。本実験では、鮮明化機器にINFOTECH社のRed Super Eye G2[15]を使用する。この鮮明化機器は、濃霧や雨などの悪天候映像、夜間や水中などの光量不足映像、西日やヘッドライト等の逆光映像といった悪条件下に撮影された映像を鮮明化することができる。映像の状態を自動で把握して濃度ヒストグラムの平均化処理や色情報処理を行うため、複雑なパラメータ設定が不要な点の特徴である。

実験結果例を以下の図11に示す。さらに、鮮明化機器および画風変換手法による鮮明画像に対して、物体検出を行った結果例を図12に示す。これらは前章同様、画風変換手法による鮮明化のほうが、検出精度が高くなる例である。



入力 HAZE 画像 鮮明化機器による鮮明化結果 画風変換手法による鮮明化結果

図11 鮮明化機器と画風変換手法の比較実験結果例

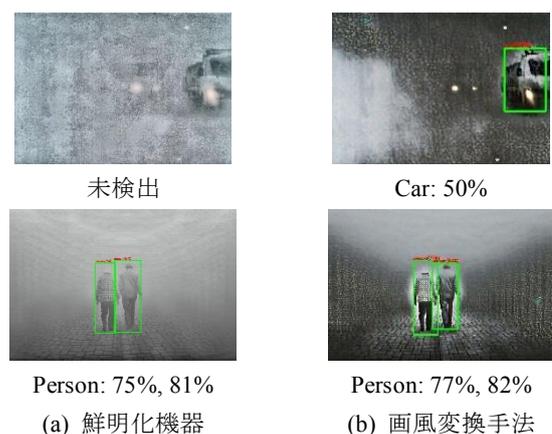


図12 物体検出結果

7. まとめと今後の課題

本稿では、画風変換を活用した HAZE 除去手法を提案した。HAZE が強い場合においては、He らによる一般的な HAZE 除去手法や鮮明化機器と比較して、より鮮明化効果が得られることを確認した。さらに、画風変換を用いた鮮明化により、物体検出精度が向上する例も確認した。

今後の課題として、HAZE 画像に写る人物や物体の検出精度を定量的に評価することで、本手法の有効性を確認すること等が挙げられる。

謝辞 本研究成果は、NICT 委託研究「ソーシャルビッグデータ利活用・基盤技術の研究開発」により得られている。

参考文献

- [1] R.Tan, Visibility in Bad Weather from a Single Image, CVPR 2008, p.1-8, 2008.
- [2] R.Fattal, Single Image Dehazing, SIGGRAPH 2008, p.1-9, 2008.
- [3] Kaiming He et al.. Single Image Haze Removal using Dark Channel Prior., CVPR2009.
- [4] Zhiming Tan et al.. Fast Single-image Defogging, FUJITSU Sci. Tech. J., Vol.50, No.1, 2014.
- [5] Leon A. Gatys et al.. A Neural Algorithm of Artistic Style, CVPR2015.
- [6] “Chainer-gogh”. <https://github.com/mattya/chainer-gogh/tree/4aaa123e839083cceb70c74d4815696ad9cfc44f>, (参照 2017-10-20).
- [7] Justin Johnson et al.. Perceptual Losses for Real-Time Style Transfer and Super-Resolution, ECCV2016.
- [8] “Chainer-fast-neuralstyle”. <https://github.com/yusuketomoto/chainer-fast-neuralstyle>, (参照 2017-10-20).
- [9] “Style Transfer in Real-Time”.<https://shafeentejani.github.io/2017-01-03/fast-style-transfer/>, (参照 2017-10-20).
- [10] “COCO-Common Objects in Context”. <http://cocodataset.org/#home>, (参照 2017-10-20).
- [11] “視界不良-1”. <http://tenkenbiyori.blog.fc2.com/blog-entry-194.html>, (参照 2017-10-21).
- [12] “吹雪視線情報”. <http://www.hokkaidolikers.com/tw/articles/3279>, (参照 2017-10-21).
- [13] Joseph Redmon et al..YOLO9000:Better,Faster,Stronger, CVPR 2016.
- [14] “YOLOv2”. <https://github.com/leetcnki/YOLOv2>, (参照 2017-10-22).
- [15] “INFOTECH-Red Super Eye G2”. http://www.infotech-japan.co.jp/rse2_3ghd_s.html, (参照 2017-10-23)