

CNN を用いた画像鮮明化システムの実装

2018SC013 林 亮佑 2018SC004 大極 康平

指導教員：河野 浩之

1 画像鮮明化が必要とされる場面

現在、製品の検査や監視カメラによる安全確保、ITS 交通システムによる道路管理などで画像が用いられている。一例として監視カメラによる安全確保を見ると、表 1 の監視カメラの画像による被疑者特定の端緒では、2020 年に画像が刑法犯数の約 10%において手がかりとなっている。しかし、画像は天候や照明条件により常に鮮明ではないので手がかりとして使用できない場合もある。ここで画像鮮明化システムを使用することで手がかりとして使用できる画像を増やすことが可能である。これと同様に、製品の検査などでも画像が不鮮明なことによる問題を画像鮮明化システムにより解決できる。

表 1: 監視カメラの画像による被疑者特定の端緒 [1]

年	刑法犯数	画像が手がかりになった件数
2018 年	36,963 件	1,401 件
2019 年	33,779 件	1,531 件
2020 年	33,009 件	3,493 件

2 低照度画像鮮明化の提案手法

2.1 で先行研究の課題と低照度画像鮮明化手法の提案、2.2 で低照度補正ネットワーク、2.3 で色改善ネットワークについて述べる。

2.1 先行研究の課題と低照度画像鮮明化手法の提案

Shen ら [2] の MSR-net では晴れた空のような非常に滑らかな領域ではハロー効果の影響を受けるといった課題があり、Huang ら [3] のネットワークでは出力画像に影が見られたり、出力画像の全体的な明るさが最適状態に達していないといった課題がある。つまり、先行研究の問題は画像の色を適切に復元することができていないことである。本研究ではこの問題を解決するために、暗い画像を明るくする低照度補正ネットワークと画像の色を適切に復元する色改善ネットワークを合わせた低照度改善ネットワークを提案する。提案手法では、色改善ネットワークにより先行研究の問題であった色の問題を解決することを狙う。

提案するネットワークを図 1 に示す。提案するネットワークでは、はじめに低照度画像を低照度補正ネットワークに入力し、低照度画像を明るくした画像である低照度補正画像を取得する。次に取得した低照度補正結果画像を色改善ネットワークに入力し、低照度補正結果画像を色改善した結果である色改善結果画像を取得し、これを最終結果とする。提案するネットワークは、Python の深層学習フ

レームワークを使用して構築し、入力画像に低照度画像、正解画像に通常照度画像を使用して学習を行う。

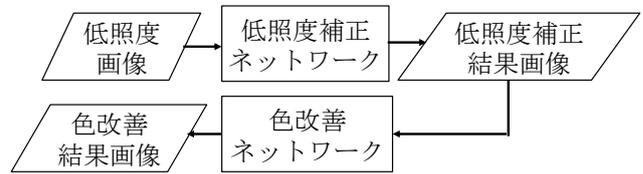


図 1: 提案する低照度鮮明化ネットワーク

2.2 低照度補正ネットワーク

低照度補正ネットワークを図 2 に示す。低照度補正ネットワークでは、1. 照明光成分予測、2. 反射成分推定、3. 反射率成分改善の 3 ステップで低照度画像を明るくする。照明光成分予測では、畳み込みオートエンコーダによって、入力された低照度画像から照明光成分を予測する。反射率成分推定では、レティネックス理論 [4] に基づいて、予測した照明光成分と入力された低照度画像から反射率成分を推定する。反射率改善成分では、複数の畳み込みにより、推定した反射率成分をより明るくする。

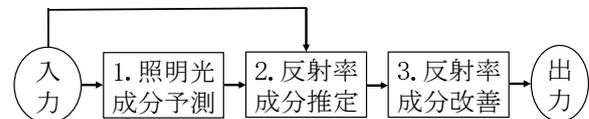


図 2: 低照度補正ネットワーク

2.3 色改善ネットワーク

色改善ネットワークの図 3 に示す。色改善ネットワークでは、畳み込みの回数を多くすることで深いネットワークを実現し、損失関数にコサイン類似度や色差などの画像間の色の類似に関する指標を使用し学習させることで色改善を行えるようにする。しかし、ただ畳み込み層を増やしてネットワークを作成するだけでは、勾配消失問題が発生してしまう。よって、色改善ネットワークにおいて深いネットワークを実現させる際にはスキップ接続を使用することで勾配消失問題を回避する。

3 低照度画像鮮明化の実験環境

3.1 でフレームワークの比較と選定、3.2 で使用するデータセットについて述べる。

3.1 フレームワークの比較と選定

本研究ではネットワークの構築に Tensorflow と Keras を使用する。Tensorflow, Keras の特徴を表 2 に示す。使

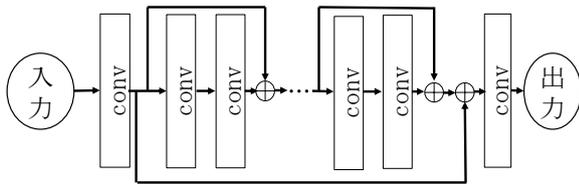


図 3: 色改善ネットワーク

用する理由として、Tensorflow は処理が高速であること、GPU での処理を行うことができることが挙げられ、Keras は Tensorflow と一緒に使用できることや高レベルの API が使用することでディープラーニングの実装が容易になることから以上の二つを使用する。

表 2: 使用するフレームワーク

フレームワーク	特徴
Tensorflow	高, 低レベルの API が使用可能 多くのプラットフォームで利用可能
Keras	ディープラーニングを比較的短い ソースコードで実装ができる

3.2 使用するデータセット

次に画像データセットの特徴を表 3 に示す。データセットに MITAdobeFiveK と LOLDataset, SICE を使用する理由として、これら 3 つは低照度画像と通常画像がペアになっている為、低照度画像の鮮明化を行うネットワークの学習に向いているからである。ダウンロードした画像は幅を 150 ピクセル、高さを 100 ピクセルにリサイズして使用する。

表 3: 使用するデータセットの特徴

データセット	画像の枚数
MIT-AdobeFiveK	5,000 組のペア画像
LOLDataset	485 組のペア画像
SICE	589 組のペア画像

さらに LOLDataset と MITAdobeFiveK, SICE の画像だけでは 6,074 枚にしか満たないためネットワークの学習には不十分なデータ量である。よって上下反転, 左右反転, 上下左右反転を通常画像と低照度画像に行い画像を生成することで, 24,296 枚にデータ量を増やす。この 24,296 枚のうち 24,000 枚を訓練用データ, 残りの 296 枚を検証用データとして使用する。

4 作成したネットワークの詳細

4.1 で作成したネットワーク全体の説明, 4.2 で畳み込みオートエンコーダを用いた低照度画像を明るくするネットワークの説明, 4.3 で色改善ネットワークの説明, 4.4 で

ネットワークの学習条件について述べる。

4.1 作成したネットワークの全体の構造

作成したネットワークの全体図を図 4 に示す。作成したネットワークは、畳み込みオートエンコーダを用いた低照度画像を明るくするネットワークと残差ユニットを用いた色改善ネットワークで構成されている。

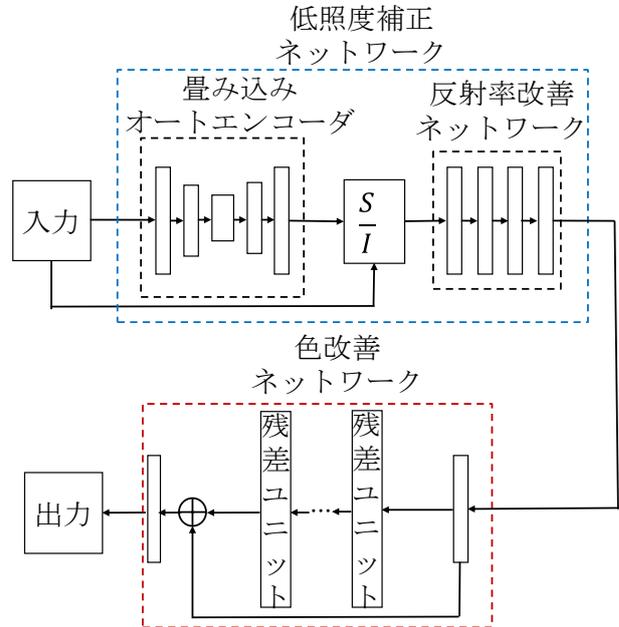


図 4: 作成したネットワークの全体像

4.2 畳み込みオートエンコーダを用いた低照度画像を明るくするネットワーク

レティネックス理論によると、画像は照明光成分と反射率成分に分解することができる。照明光成分とは被写体に照射された照明光の成分であり、反射率成分とは照明に依存しない画像成分である。そして、レティネックス理論は次のように定式化される。

$$S = I * R \quad (1)$$

S は画像を表し, I は照明光成分, R は反射率成分, $*$ はピクセル単位の乗算を表す。

畳み込みオートエンコーダのネットワーク図を図 5 に示す。この畳み込みオートエンコーダは、畳み込み層と最大プーリング層でエンコードをし、転置畳み込み層でデコードをするネットワークである。エンコードで画像のデータを圧縮し、デコードで画像を復元することでぼやけた画像を取得する。レティネックス理論では、照明光成分は低周波特性を持つ性質があり、ぼやけた画像は画像全体にわたって低周波特性を持っているので照明光成分として使用する。そして畳み込みオートエンコーダのフィルタ数 (A, B, C) を決定するために複数のパラメータで比較を行った。その結果、フィルタ数を $(A, B, C) = (32, 64, 32)$ とした。

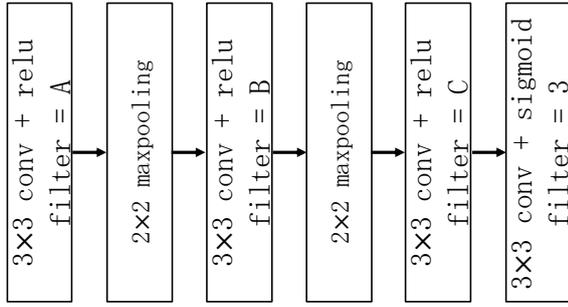


図 5: 畳み込みオートエンコーダ

反射率成分を求める処理では入力画像とオートエンコーダで取得した照明光成分を使い、式 (1) を計算することで反射率成分を取得している。

反射率成分改善ネットワークのネットワーク図を図 6 に示す。反射率成分改善ネットワークは反射率成分に対し、畳み込み層を用いることでより明るい画像を取得している。活性化関数は ReLU 関数を使用し、最後の層のみ Sigmoid 関数を使用している。さらにこのネットワークでは 2 層の畳み込み層と 1 回のバッチ正規化が 1 セットとしている。そのため、セット数を変更した時の性能を比較を行った。比較の結果、セット数を 3 に設定した。

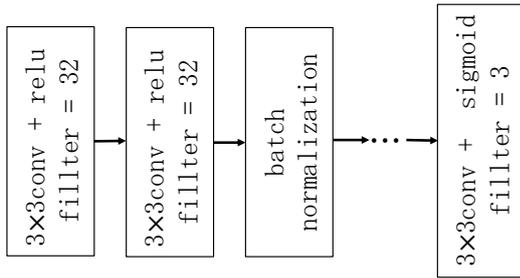


図 6: 畳み込み層による反射率成分改善ネット

4.3 色改善ネットワーク

色改善ネットワークのネットワーク図を図 7 に示す。この色改善ネットワークでは残差ユニットと呼ばれるネットワークを使用して深いネットワークを構築している。深いネットワークにして畳み込みの回数を増やすことでより色を改善を行う。

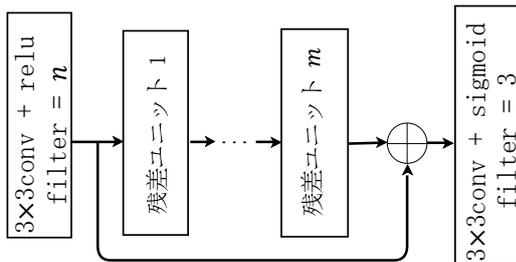


図 7: 色改善ネットワーク

図 8 に残差ユニットの構造を示す。この残差ユニットでは、ある層に与えられた信号を少し上位の層の出力に追加するスキップ接続を使用することで、層の最適な出力ではなく、層の入力と出力の差分を学習すればよくなるので、最適化がしやすくなる。これにより、深いネットワークを構築できるようになる。

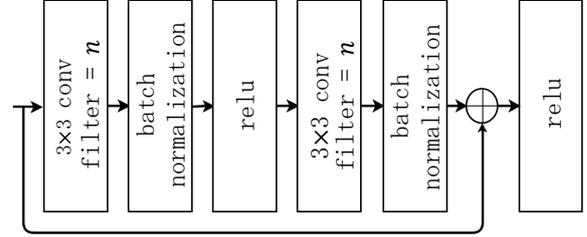


図 8: 残差ユニット

そして色改善ネットワークの最適なパラメータを決定するために残差ユニットの個数、畳み込み層のフィルタ数を変更していき比較を行った。その結果、フィルタ数 n を 8、残差ユニットの個数 m を 10 とした。

4.4 学習条件

低照度補正ネットワークの学習条件として、最適化手法は Adam、学習率は 0.001、エポック数は 270、バッチサイズは 32 とした。そして、学習に使用する損失関数には、SSIM を損失関数にしたものである $SSIM_{LOSS}$ を使用する。 $SSIM_{LOSS}$ の式を式 (2) に示す。

$$SSIM_{LOSS}(\hat{x}, x) = 1 - \frac{(2\mu_{\hat{x}}\mu_x + C_1)(2\sigma_{\hat{x}x} + C_2)}{(\mu_{\hat{x}}^2 + \mu_x^2 + C_1)(\sigma_{\hat{x}}^2 + \sigma_x^2 + C_2)} \quad (2)$$

ここで、 \hat{x} は正解画像、 x はネットワークによる予測画像、 $\mu_{\hat{x}}, \mu_x$ はそれぞれ \hat{x}, x の平均画素値、 $\sigma_{\hat{x}}, \sigma_x$ はそれぞれ \hat{x} と x の画素値の標準偏差、 $\sigma_{\hat{x}x}$ は共分散、 C_1, C_2 は定数を表す。 $SSIM_{LOSS}$ では、正解画像と予測画像の構造が一致するほど値は 0 に近づき、一致しないほど値は 1 に近づく。

色改善ネットワークの学習条件として、最適化手法は Adam、学習率は 0.001、エポック数は 180、バッチサイズは 32 とした。そして、学習に使用する損失関数には、 $SSIM_{LOSS}$ と COS_{LOSS} 、 $ColorDiff_{LOSS}$ 、 $CIQI_{LOSS}$ を合わせた $ColorImprovement_{LOSS}$ を使用した。 $ColorImprovement_{LOSS}$ では、画像の SSIM を損なわず、色改善を行うことを狙う。 $ColorImprovement_{LOSS}$ の式を式 (3)、 COS_{LOSS} の式を式 (4)、 $ColorDiff_{LOSS}$ の式を式 (5)、 $CIQI_{LOSS}$ の式を式 (6) に示す。

$$ColorImprovement_{LOSS}(\hat{x}, x) = 4 * SSIM_{LOSS}(\hat{x}, x) + 3 * COS_{LOSS}(\hat{x}, x) + 2 * ColorDiff_{LOSS}(\hat{x}, x) + CIQI_{LOSS}(x) \quad (3)$$

$$COS_{LOSS}(\hat{x}, x) = -\frac{(\hat{x}, x)}{\|\hat{x}\| \|x\|} \quad (4)$$

$$ColorDiff_{LOSS}(\hat{x}, x) = \frac{1}{H} \frac{1}{W} \sum_{i=1}^H \sum_{j=1}^W \{(\hat{R}_{ij} - R_{ij})^2 + (\hat{G}_{ij} - G_{ij})^2 + (\hat{B}_{ij} - B_{ij})^2\} \quad (5)$$

$$CIQI_{LOSS}(x) = -(\sqrt{\sigma_\alpha^2 + \sigma_\beta^2} + 0.3 * \sqrt{\mu_\alpha^2 + \mu_\beta^2}) \quad (6)$$

ここで、 \hat{x} は正解画像、 x はネットワークによる予測画像を表す。 H は画像の高さ、 W は画像の幅、 \hat{R}_{ij} , \hat{G}_{ij} , \hat{B}_{ij} は正解画像 \hat{x} の i 行 j 列番目のピクセルの各 RGB 値を表し、 R_{ij} , G_{ij} , B_{ij} はネットワークによる予測画像 x の i 行 j 列番目のピクセルの各 RGB 値を表す。 σ_α , σ_β , μ_α , μ_β は RGB 画像の各カラーチャンネル (R, G, B) を用いて $\alpha = R - G$, $\beta = 0.5(R + G) - B$ としたとき、 σ_α , σ_β は α , β の各分散、 μ_α , μ_β は α , β の各平均を表す。

COS_{LOSS} は -1 に近いほど正解画像の色と類似していることを示す。 $ColorDiff_{LOSS}$ は値が小さいほど正解画像の色と差が無いことを示す。 $CIQI_{LOSS}$ は値が大きいほど画像が鮮やかであることを示す。

5 色改善ネットワークの有無による性能比較

低照度補正ネットワーク (以下低照度補正 net とする) と提案するネットワーク (以下色改善 net ありとする) を比較することで色改善ネットワークにより画像の色が改善されているのか確認する。比較では、両ネットワークに低照度画像を入力し、その出力画像と正解画像を用いて PSNR, SSIM, コサイン類似度, 色差 (LAB), CIQI で計測を行うことで出力画像が正解画像とどの程度類似しているか確認する。PSNR は値が大きいほど画質の劣化が少ないことを示す。SSIM は値が 1 に近いほど画像の構造が類似していることを示す。コサイン類似度は値が 1 に近いほど画像の色が類似していることを示す。色差は値が小さいほど画像の色が類似していることを示す。CIQI は値が高いほど画像の色が鮮やかであることを示す。ここで使用する正解画像は Google 検索で入手した画像 28 枚と現実世界で撮影した通常照度画像 2 枚であり、低照度画像は Google 検索で入手した画像に対してガンマ補正を行った画像 28 枚と現実世界で撮影した低照度画像 2 枚である。

評価値ごとの平均値を表 4, 出力画像の一部を図 9 に示す。表より色改善ネットワークは低照度補正ネットワークと比較して PSNR は 1.04 %, SSIM は 0.31 %, コサイン類似度は 0.15 %, 色差は 0.15 %, CIQI は 23.4 % 向上している。コサイン類似度, 色差, CIQI は色の評価値であり, これらすべてが向上していることから色改善ネットにより画像の色を改善していることが確認できた。また, 出力画像では低照度が改善されており, さらに Shen らの課題であったハロー効果などの歪みも見られなかった。

表 4: 低照度補正 net と色改善 net ありの評価値

評価指標	低照度補正 net	色改善 net あり
PSNR	16.7055	16.8817
SSIM	0.79640	0.79884
コサイン類似度	0.97012	0.97157
色差 (LAB)	17.8805	17.8529
CIQI	11.9402	15.6003



図 9: 低照度補正 net と色改善 net ありの出力画像例

6 結び

低照度補正ネットワークと色改善ネットワークを組み合わせた低照度画像鮮明化システムを提案した。これにより, 低照度画像を明るくし, 色改善を行うことができた。さらに, 出力画像ではハロー効果のような歪みがない画像を取得できた。

一方, 比較対象となる MSR-net の実装を行ったが, 提案手法と正しく比較することができなかった。

参考文献

- [1] 警視庁, “警視庁の統計,” https://www.keishicho.metro.tokyo.lg.jp/about_mpd/jokyo_tokei/tokei/, 参照 Oct. 2021.
- [2] Liang Shen, Zihan Yue, Fan Feng, Quan Chen, Shihao Liu, Jie Ma, “MSR-net: Low-light Image Enhancement Using Deep Convolutional Network,” arXiv:1711.02488v1 [cs.CV], Nov. 2017.
- [3] Wei Huang, Yifeng Zhu, Rui Huang, “Low Light Image Enhancement Network With Attention Mechanism and Retinex Model,” IEEE Access, Vol.8, pp. 74306-74314, Apr. 2020.
- [4] Edwin Land, John McCann, “Lightness and Retinex Theory,” Journal of the Optical Society of America, Vol.61, No. 1, pp. 1-11, Jan. 1971.