

# RealSense カメラを用いた機械学習による深度推論の検証

S19096 小島 龍輝

## 1. はじめに

これまでの研究では、Intel RealSenseD435 カメラで撮影したRGB画像と深度画像(RGB-D画像)をペアとして作成したデータセットを用い、GAN(敵対的生成ネットワーク)の一種であるPix2Pixライブラリを使用した学習及び推論を行った。

本研究では、これまでの研究で課題となった推論精度に関して着目し、これを向上させる手法について掘り下げ、混合精度を用いたミニバッチサイズ拡充による精度向上の取り組みを行った。そして、これら学習にあたりデータセットの拡充・機械学習ネットワークの変更・学習した推論データの検証を行った。

## 2. データセットの作成

図1は作成したデータセットの例である。これまで160枚の画像ペアで学習を行っていたが、本研究では、これを拡充し456枚の追加撮影を行った。

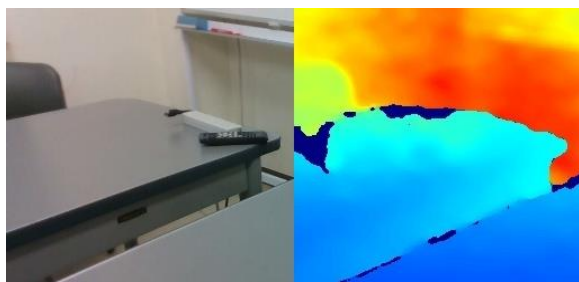


図1 作成したデータセットの例

撮影した画像サイズが640\*360ピクセルで、学習に用いる画像サイズが256\*256ピクセルであることに着目し、元の画像から256\*256サイズの画像の切り抜きを行い、データセットの水増しを行っ

た。この際画像の切り出し位置を50ピクセルずつずらすことで、1枚の画像から24枚のデータセットを作成した。

## 3. Pix2Pixによる学習結果

これまでの研究はTensorflowによるPix2Pixライブラリを使用していたが、長時間の学習を行う過程でエラーが多発する現象が起きてしまい、データ量を増やした学習が困難であった。そのため、本研究ではPix2PixをPythonのPytorch機械学習ライブラリを用いて実装し、これを活用した学習を行うことにした。

図2は、ミニバッチサイズ32で学習を行った結果である。図2内でCross Entropyと示したのはBCE: Binary Cross Entropyのことである。ここに注目すると学習の進行に応じて生成器(Generator)の損失が上下する部分があるものの、山の後の損失の差が少なく、300エポックの学習で収束したと判断できる。しかし、エポックの進捗に対しL1 lossの増加傾向が見られ、正常な学習が行えていないものと考えられる。

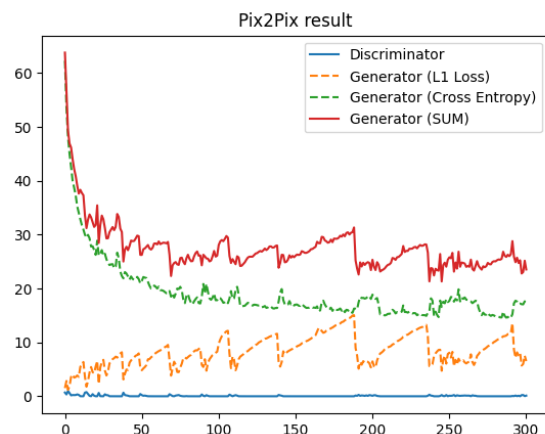


図2 Pix2Pix Batch32 学習ログ

## 4. 混合精度を活用した学習結果

Batch32 での学習時に問題となった L1 loss の増加傾向について、ミニバッチサイズを 64 へと拡張し、学習データからの学習率を変更することにより精度が向上するのではないかと考えた。混合精度という手法<sup>2)</sup>を用いて GPU 上のメモリサイズを半減させ、ミニバッチサイズを 2 倍の 64 で学習を行った。

図 3 は、ミニバッチサイズ 64 で学習を行った結果である。学習曲線において、ミニバッチ 32 での学習と比べるとミニバッチ 64 の方が学習率はなだらかであるが、最終的なロス値は少なくなった。

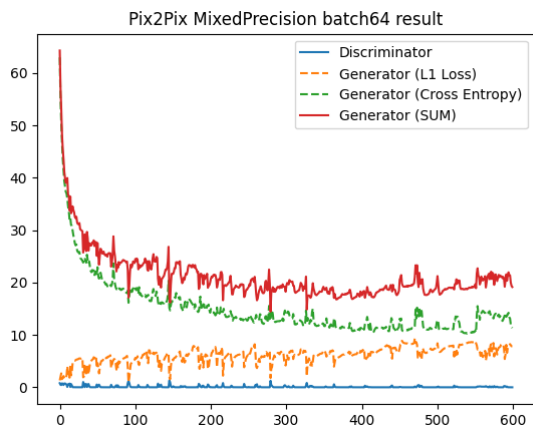


図 3 Pix2Pix Batch64 学習ログ

## 5. 推論比較

図 4 はミニバッチ 32 および 64 で学習させたモデルに動画を推論させ、その 1 フレームを抜粋したものである。B はバッチサイズ、E はエポック数を表し、ミニバッチサイズ 64 については学習後半の 100 エポック刻みでサンプリングを行った。

各推論後の画像を見ると、バッチ 32、エポック 199 の推論データが最も元画像に近い推論を行っていることが分かる。これは意外な結果であった。表 1 は、動画の 1 フレームを抜粋して L1 loss (MAE) を算出したものである。これにより、学習曲線と実際の推論精度は一致しないことがわかった。

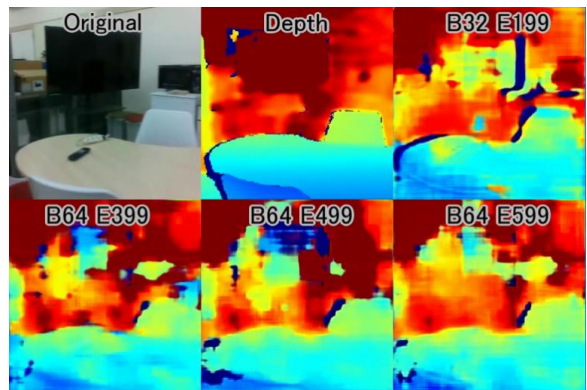


図 4 推論結果

表 1 検証データの L1 loss(MAE)

エポック数	99	199	299	399	499	599
ミニバッチ数32	34.208	30.398				
ミニバッチ数64	43.333	39.793	35.032	44.234	39.496	39.053

## 6. おわりに

本研究の結果として、画像内の近傍の特徴的な形状を捉えることには成功した。しかしながら、遠方の形状については学習不足と考えられる着色のブレが多く見受けられる結果となった。これは Real Sense カメラの精度の不足と、低解像度画像を用いたことによるものと考えられる。

また、今回はミニバッチを拡張することによる精度向上の試みには失敗してしまったが、この結果は、各パラメータを変更する際や他手法を適用する際の参考点としたい。

## 7. 参考文献

- 1) pix2pix を 1 から実装して白黒画像をカラー化してみた (PyTorch)  
[https://blog.shikoan.com/pytorch\\_pix2pix\\_colorization/](https://blog.shikoan.com/pytorch_pix2pix_colorization/) (2023.1.20 参照)
- 2) 廣川 雄一, 大西 領, 安田 勇輝, Kolomenskiy Dmitry, 杉山 大祐, “微気象シミュレーション超解像の浮動小数点精度に関する検討” 第 35 回数値流体力学シンポジウム E01-5