

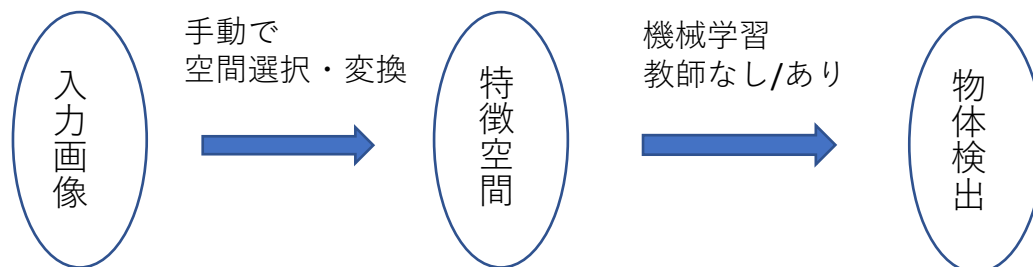
ガウス球へのドメイン変換 による教師なし物体検出

2022/6/30

@samacoba

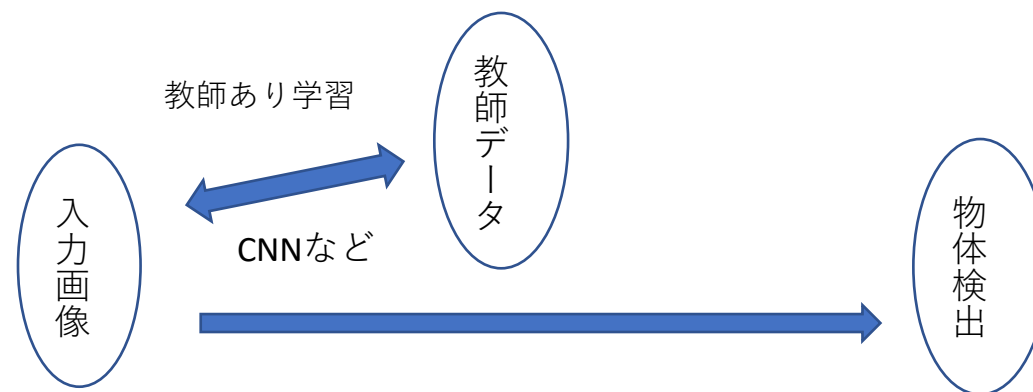
研究背景

・ 前処理
+ 機械学習



前処理に
人の労力が必要

・ 教師あり
深層学習



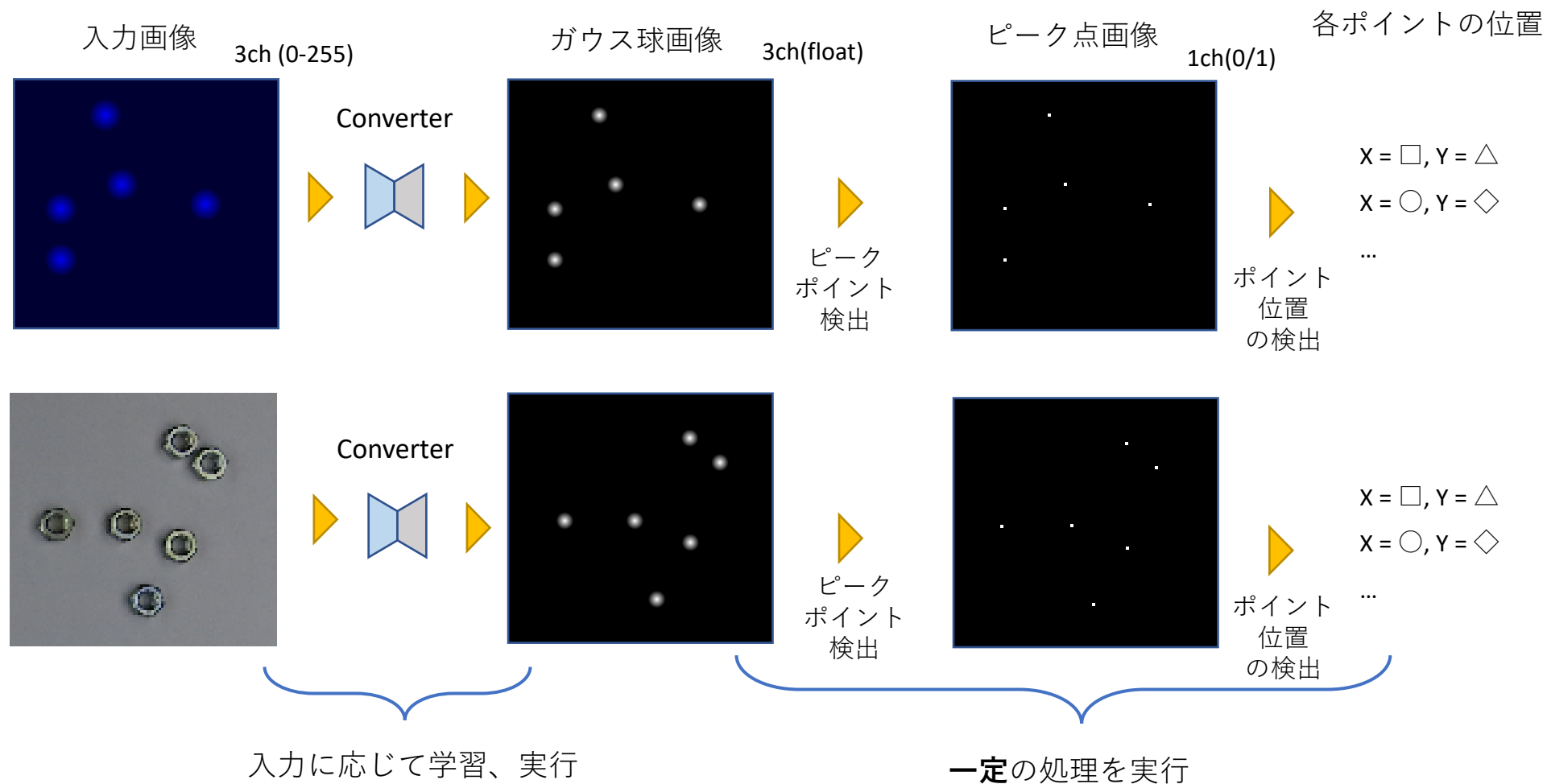
教師データ作成に
人の労力が必要

研究目標

前処理なし・教師なしで、 多様なオブジェクトの検出・クラス分けを目指す

↑
人間が物体に合わせて
特徴選択や抽出をしない

↑
色・テクスチャ・形状にかかわらず



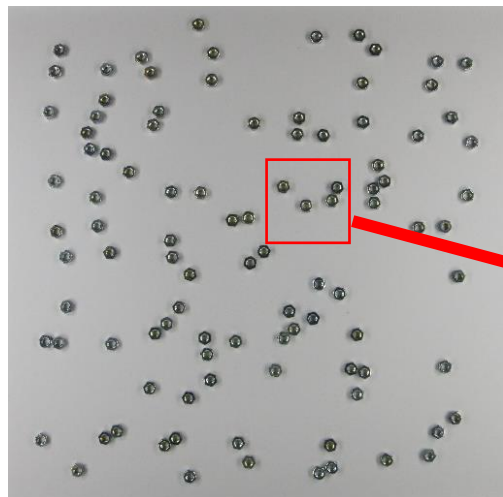
入力に応じて「Converter」が自動で学習できれば、一定の処理で検出が可能

本研究では一度検出しやすいガウス球へ変換し、その後検出を行う

提案手法

・学習時の入力データ

入力画像A



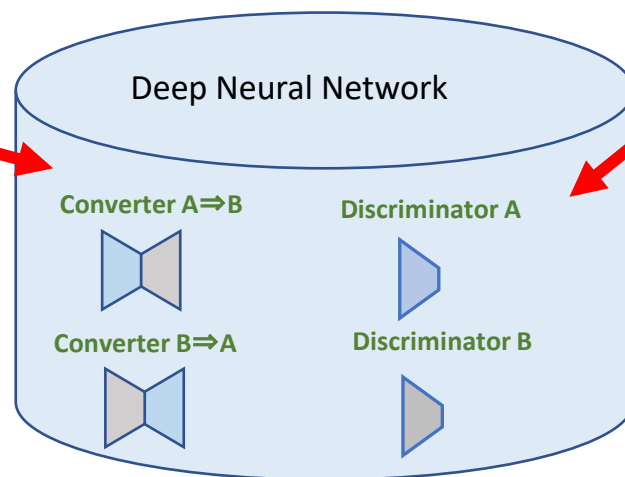
$3\text{ch} \times 512 \times 512 \times 1$ 枚

ランダムガウス球画像B



$3\text{ch} \times 512 \times 512 \times 10$ 枚

位置は一致していない



入力画像と**同オーダ**の数のガウス球をランダムな位置に散在させて**生成する**

※ 与えるガウス球の平均の**散布数**は重要なパラメータ

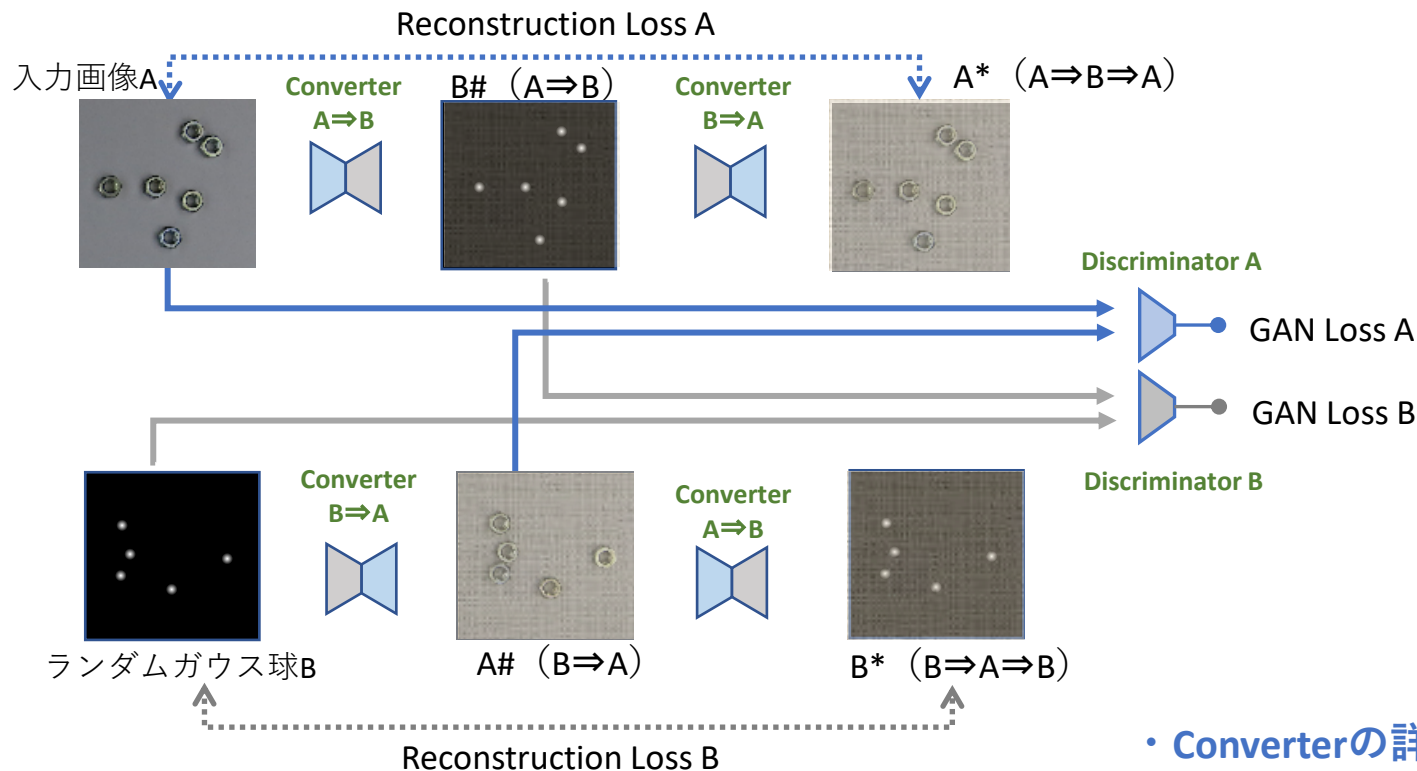
「**ヒント**」を与えているため、完全な教師なしではない

入力画像Aとランダムガウス球画像Bを $64 \times 64 \times 100$ 枚ずつ切り取り、

4種類のDeep Neural Networkを学習させる

提案手法

• Converterの学習



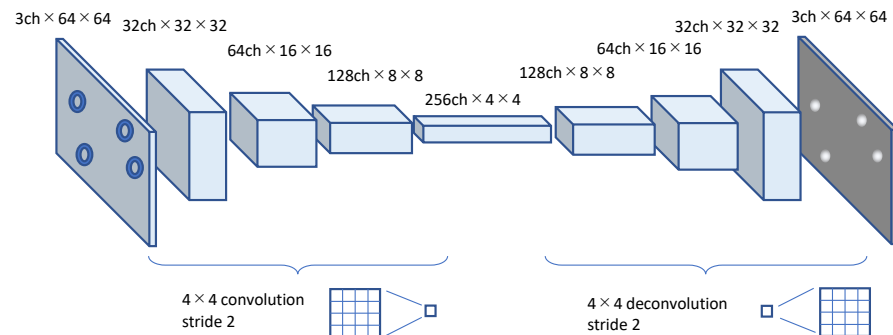
DiscoGAN[1]のドメイン変換学習ネットワークを転用



[1]Taeksoo Kim, et al. "Learning to Discover Cross-Domain Relations with Generative Adversarial Networks", ICML(2017)

- DiscoGANのネットワークを転用し、ランダムガウス球を使って学習
- 4つのLossで、2つのConverterと2つのDiscriminatorを学習

• Converterの詳細



1class 検出

サンプル（1class用）

100個



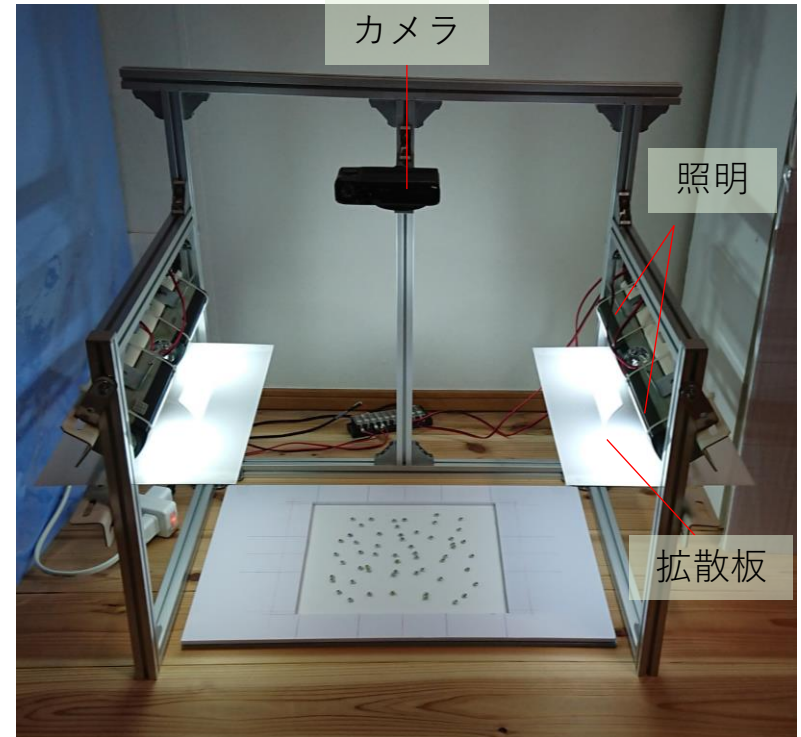
50個



25個



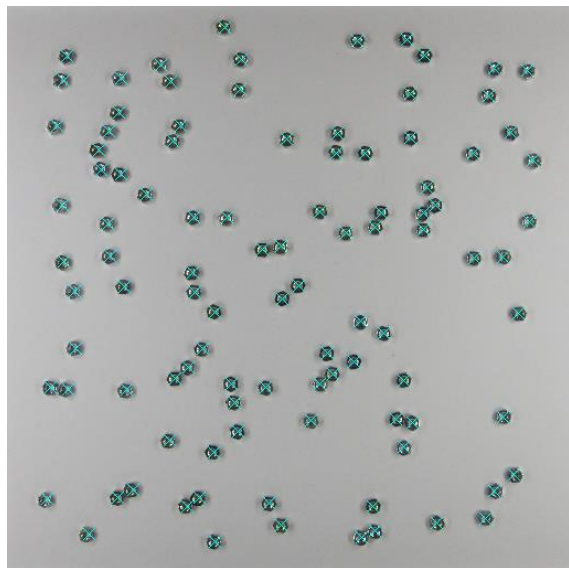
撮影環境



- ・ M3ナットを100個、50個、25個配置し、各5枚ずつ撮影
- ・ サンプルは撮影毎に「手」でランダム風に置きなおしている

学習過程

入力A 正解100個
(出力位置 ×)



出力B # 100個
(iter = 2000)



ランダムガウス球 B
(散布数 平均100個)



出力B # 15個
(iter = 300)



出力B # 90個
(iter = 600)

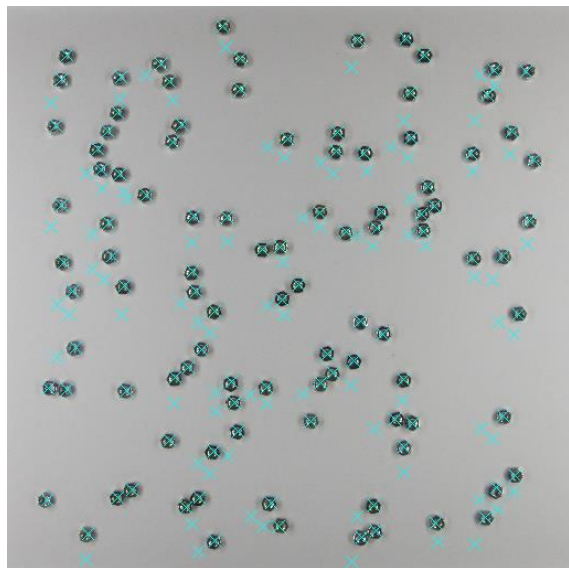


出力B # 98個
(iter = 900)



失敗例

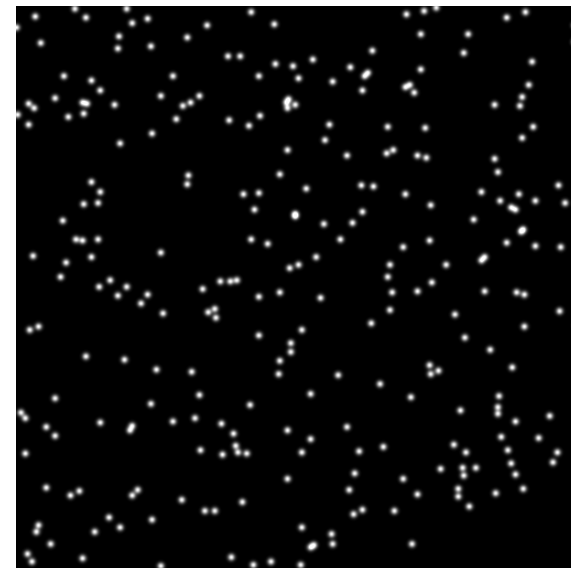
入力A 正解100個
(出力位置 ×)



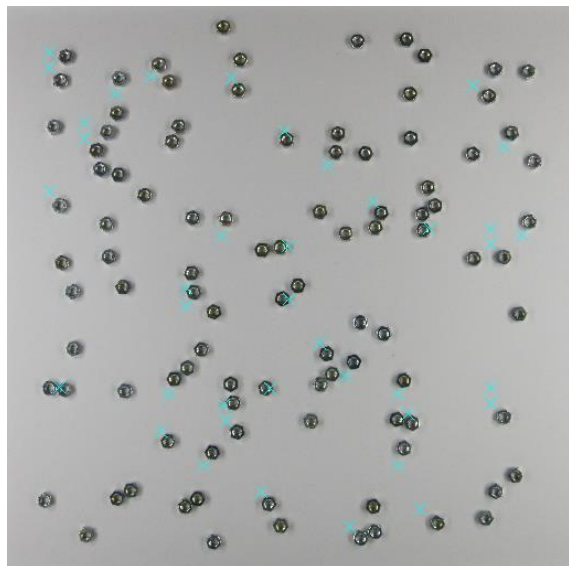
出力B # 178個
(iter = 2000)



ランダムガウス球 B
(散布数 平均**300**個)



入力A 正解100個
(出力位置 ×)



出力B # 39個
(iter = 2000)



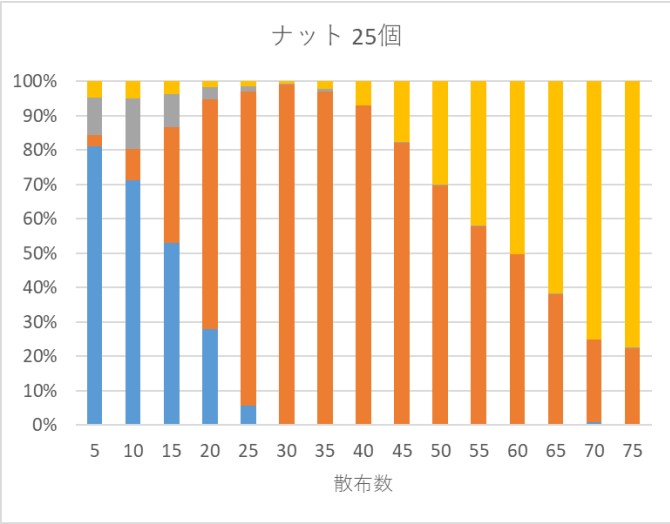
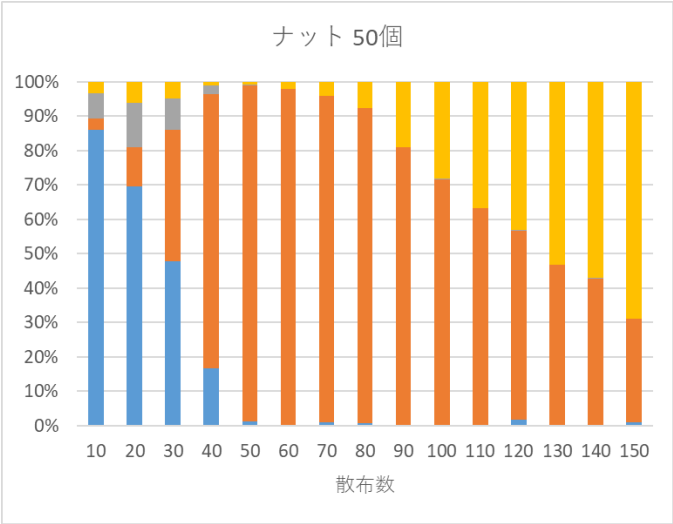
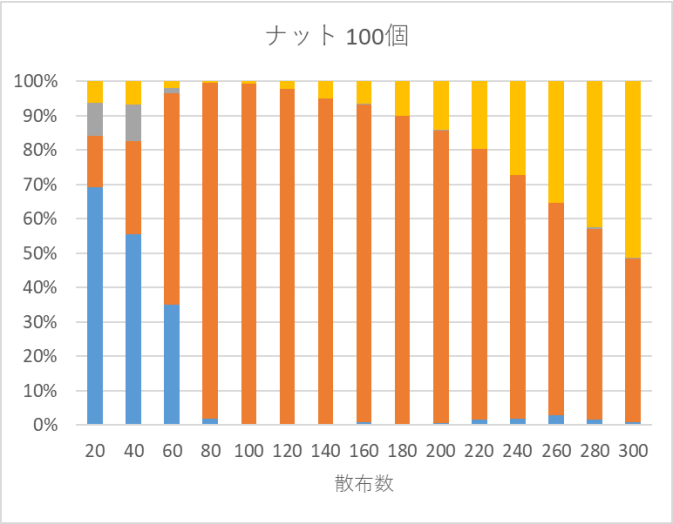
ランダムガウス球 B
(散布数 平均**40**個)



散布数と出力数

・ 100個、50個、25個のナットが写った画像に対し、散布数を変化させてOK数を評価

各条件にて 画像5枚×20回学習 = 100回平均

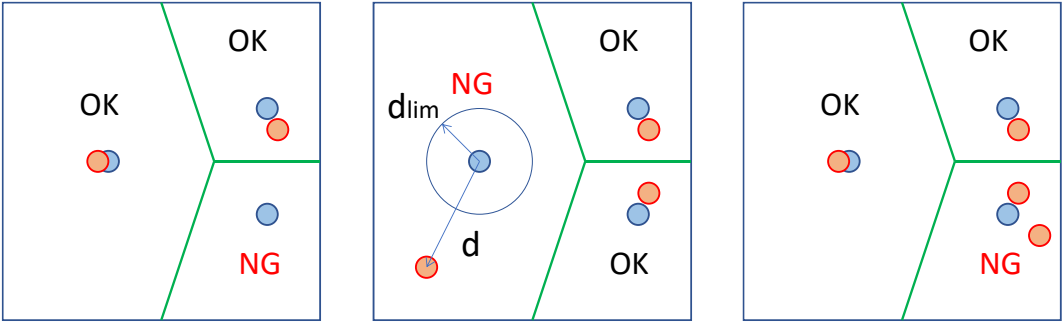


■ ゼロ ■ OK ■ 範囲外 ■ 2 個以上

評価方法

dlim :8pixel以内ならOKとした

- ・ 散布数が正解数より小さいとゼロが増える
- ・ 散布数が正解数の2倍を超えると出力過剰増える



ゼロ

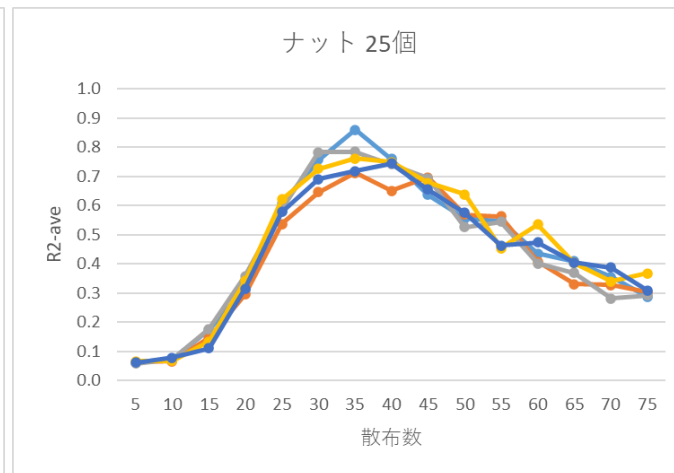
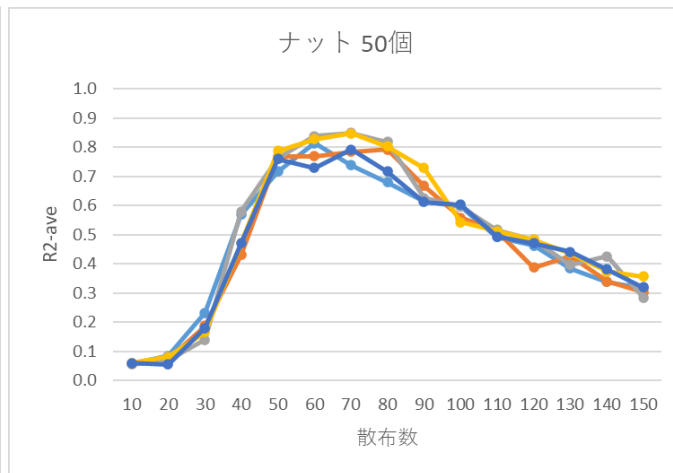
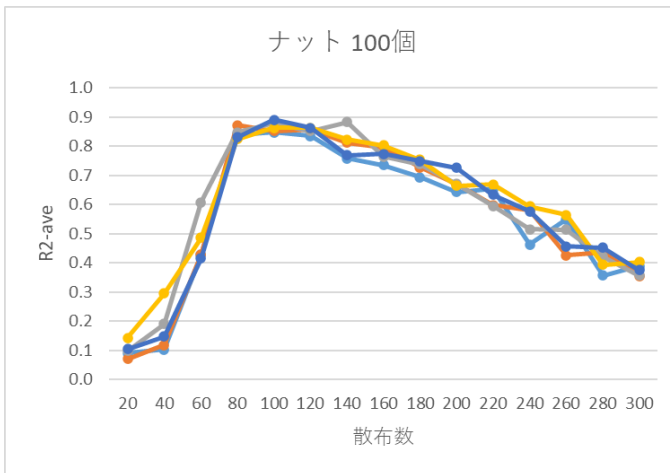
範囲外

2個以上

●: 正解位置 ●: 出力位置

・ 散布数とR2

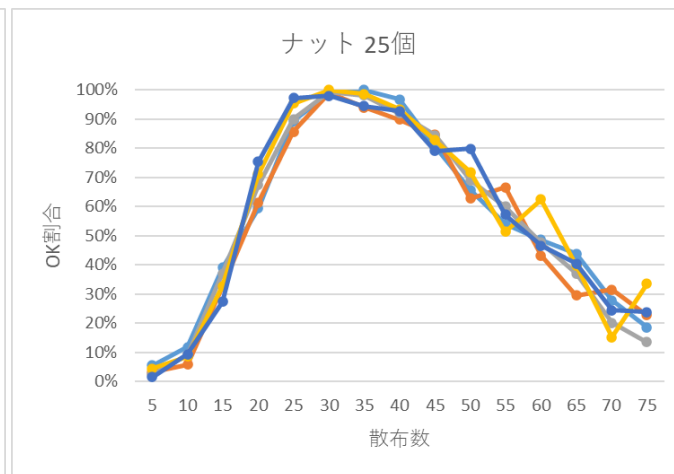
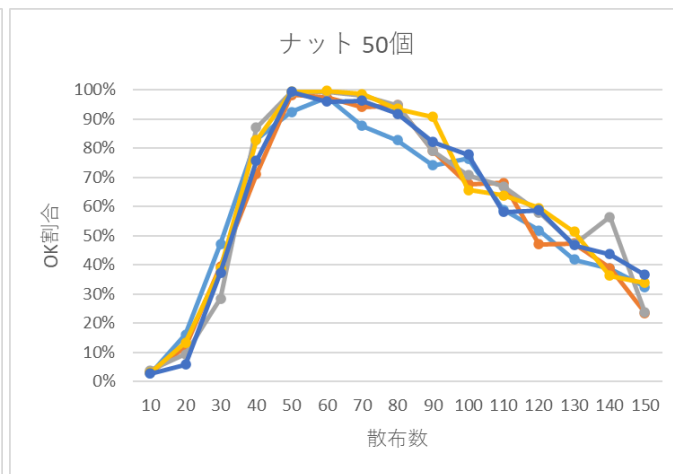
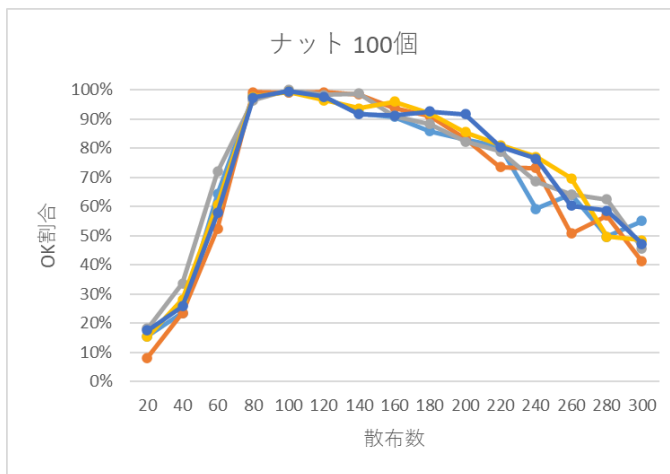
出力画像と出力の平均画像(20枚)に対し、R2を計算（正解データは使っていない）



img0 img1 img2 img3 img4

・ 散布数とOK割合

出力画像と正解データを使ってOKとなった割合を計算



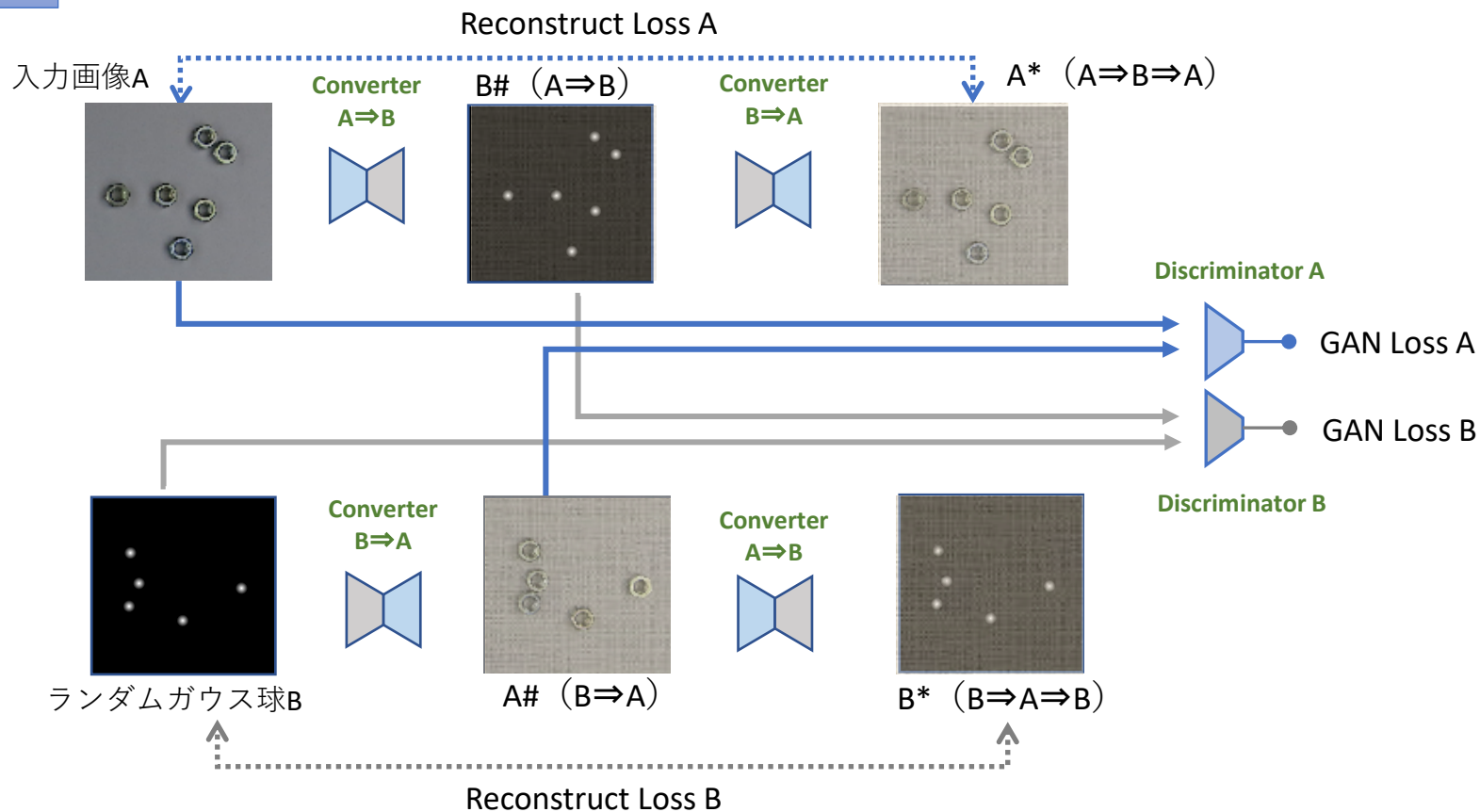
img0 img1 img2 img3 img4

R2のピークとOK割合のピークがほぼ一致しており、正解データがない状態でもR2により最適な散布数が推定できる可能性がある

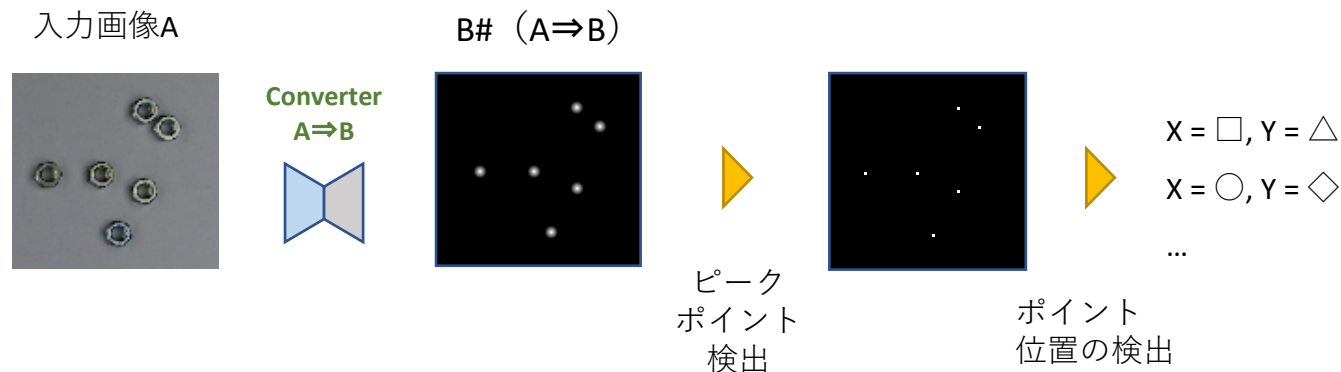
教師有無の比較

教師なし

学習時



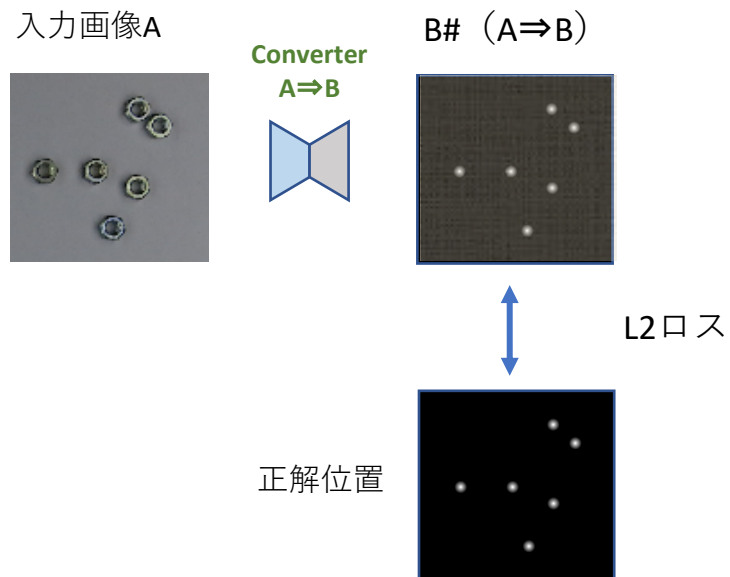
テスト時



教師有無の比較

教師あり

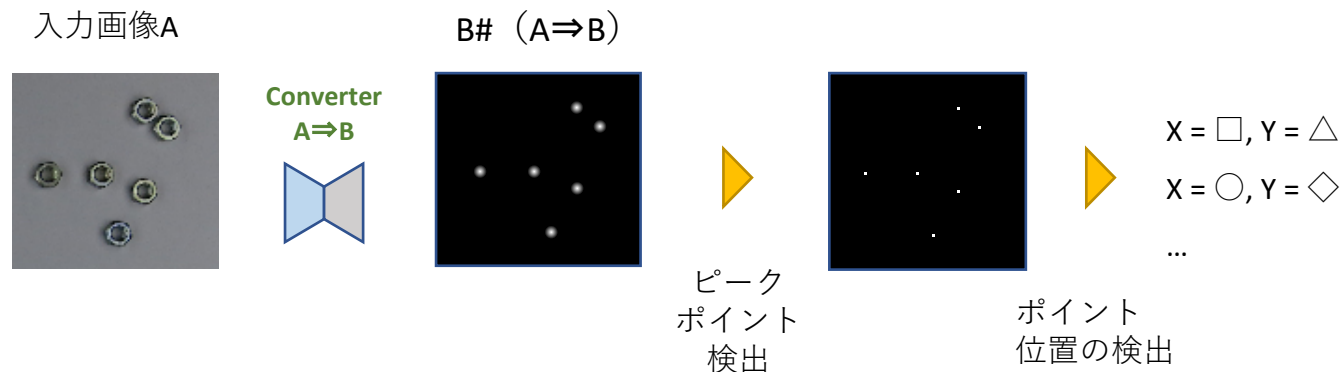
学習時



・ 直接ガウス球を正解位置に配置した画像を教師としてL2で学習

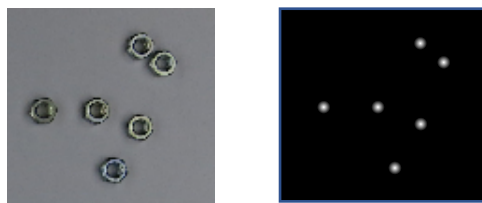
・ 同一のネットワーク構造にて、 $A \Rightarrow B$ Converterの学習が教師ありに比べて教師なしでどの程度うまくいくかを比較する

テスト時

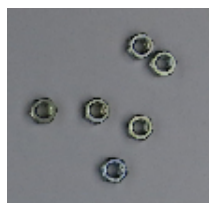


教師有無の比較

教師あり



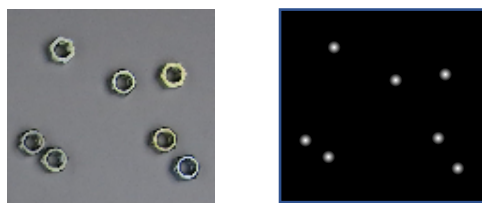
img0で学習



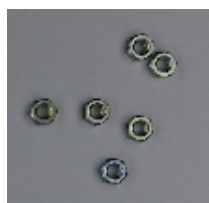
img0で評価

簡単

教師あり

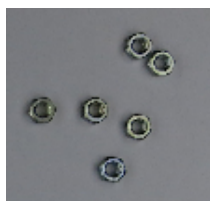


img1で学習

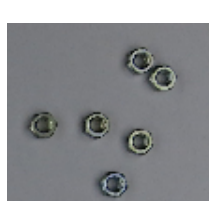


img0で評価

教師なし

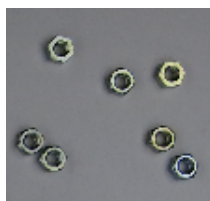


img0で学習

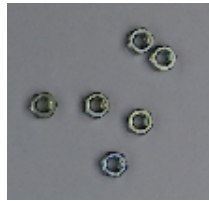


img0で評価

教師なし



img1で学習



img0で評価

難しい

教師あり

評価画像

画像
類似度

	0	1	2	3	4
0					
1					
2					
3					
4					

教師なし

評価画像

画像
類似度

	0	1	2	3	4
0					
1					
2					
3					
4					

5画像(img0,1,2,3,4) ×
各20回計算

教師有無の比較

- ・ ナット100個の画像5枚、各20回学習
- ・ 教師なしの散布数は100個
- ・ 実際100個中OKの個数（20回平均）

教師あり

		評価画像						評価平均	
		img0	img1	img2	img3	img4	5枚平均	学習=評価	学習≠評価
学習画像	img0	99.80	99.50	99.50	99.85	99.85	99.70	99.80	99.68
	img1	99.95	100.00	100.00	100.00	100.00	99.99	100.00	99.99
	img2	99.85	99.90	99.85	99.95	99.85	99.88	99.85	99.89
	img3	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
	img4	99.15	99.45	99.20	98.25	99.20	99.05	99.20	99.01
	5枚平均	99.75	99.77	99.71	99.61	99.78	99.72	99.77	99.71

教師なし

		評価画像						評価平均	
		img0	img1	img2	img3	img4	5枚平均	学習=評価	学習≠評価
学習画像	img0	99.80	99.50	99.50	99.85	99.85	99.70	99.80	99.68
	img1	99.75	99.80	99.80	99.85	99.60	99.76	99.80	99.75
	img2	99.05	99.10	98.85	98.70	98.90	98.92	98.85	98.94
	img3	99.85	99.85	99.90	99.90	99.65	99.83	99.90	99.81
	img4	99.50	99.30	99.30	99.35	99.45	99.38	99.45	99.36
	5枚平均	99.59	99.51	99.47	99.53	99.49	99.52	99.56	99.51

教師なしは教師ありに近いレベルでモデルが学習できており、
学習=評価と学習≠評価の差は小さい

Lossの影響度

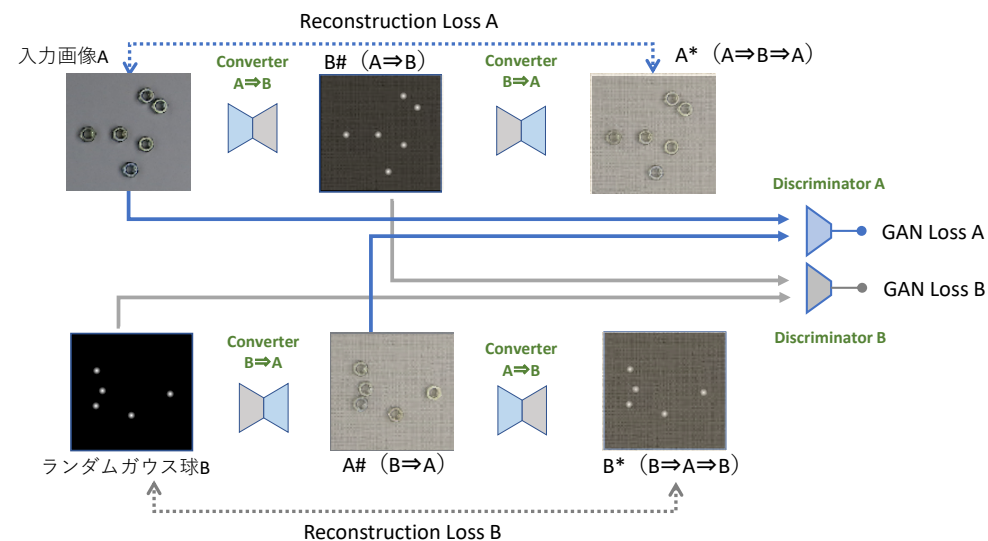
ネットワークのLossの一部をゼロにし、影響を調べる

- ・ ナット100個の画像5枚×20回 = 100回
- ・ 散布数100個

Lossの有無					評価	
Loss数	Recon A	Recon B	GAN A	GAN B	平均OK割合	OK※90%以上
4	○	○	○	○	99.7%	100枚
3	×	○	○	○	96.6%	96枚
	○	×	○	○	13.8%	6枚
	○	○	×	○	100.0%	100枚
	○	○	○	×	42.4%	0枚
2	○	○	×	×	34.0%	0枚
	○	×	○	×	0.0%	0枚
	○	×	×	○	67.2%	66枚
	×	○	○	×	21.0%	0枚
	×	○	×	○	82.2%	77枚
	×	×	○	○	11.6%	0枚
1	○	×	×	×	0.0%	0枚
	×	○	×	×	5.9%	0枚
	×	×	○	×	0.0%	0枚
	×	×	×	○	11.9%	0枚
○/成功	3/5	4/5	2/5	5/5		

※100回中OKが90%以上の枚数

Converterの学習



- ・ GAN Bは必須、GAN Aは必須ではない
- ・ Recon AとRecon Bのどちらかが必要、ただし影響度はRecon B > Recon A

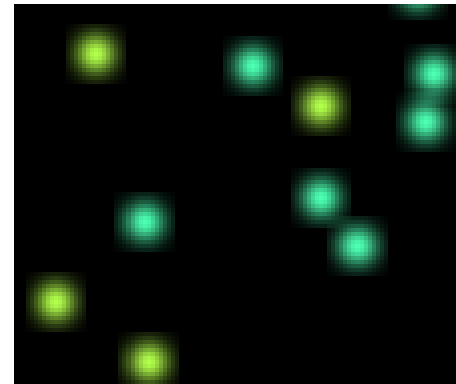
2 class 検出

サンプル (2class用)

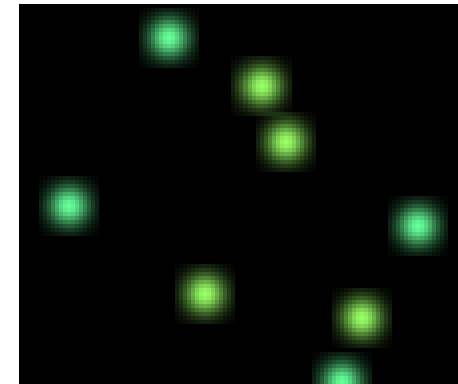
- YG/BG球 [7:3] (CG生成)
黄緑球(R:G:B = 7:10:3) / 青緑球(R:G:B = 3:10:7)
平均50個 / 平均50個
- YG/BG球 [6:4] (CG生成)
黄緑球(R:G:B = 6:10:4) / 青緑球(R:G:B = 4:10:6)
平均50個 / 平均50個
- mesh 1pix/2pix (CG生成)
1ピクセル毎 / 2ピクセル毎 のパターン
平均50個 / 平均50個
- beads 青/赤 (写真)
青ビーズ / 赤ビーズ
50個 / 50個
- nut / washer (写真)
M3ナット / M3ワッシャー
50個 / 50個

各条件にてCG生成は100枚×1回、写真は5枚×20回

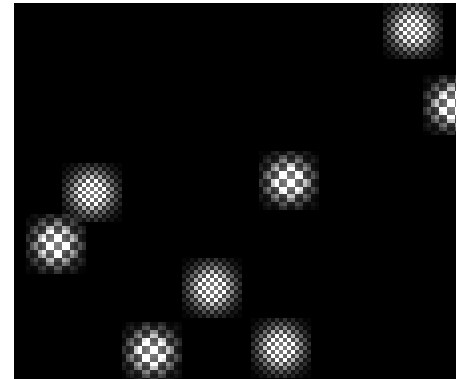
YG/BG球 [7:3]



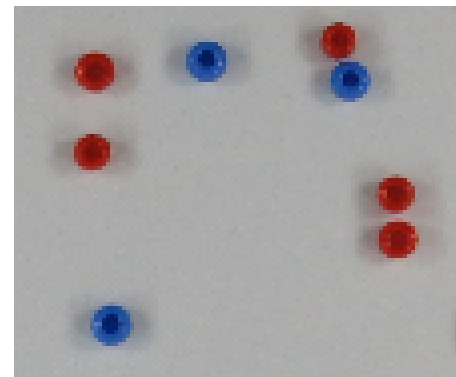
YG/BG球 [6:4]



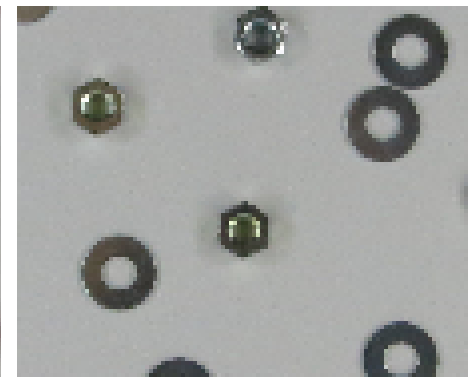
mesh 1pix/2pix



beads 青/赤

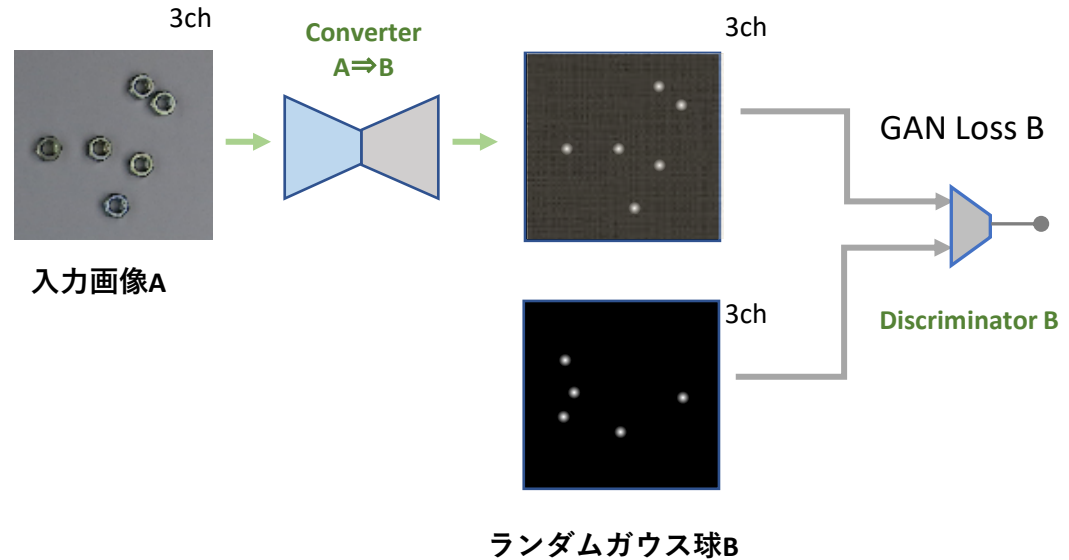


nut / washer

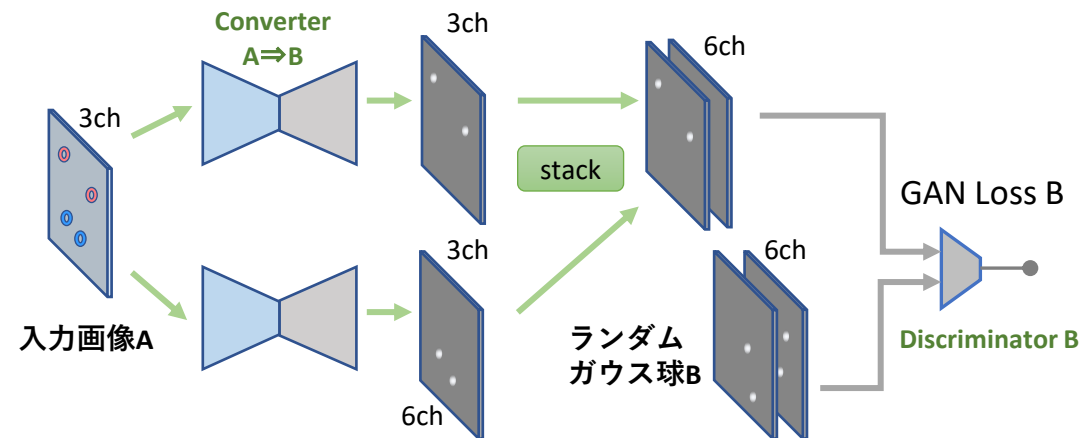


1class検出と2class検出

1class検出



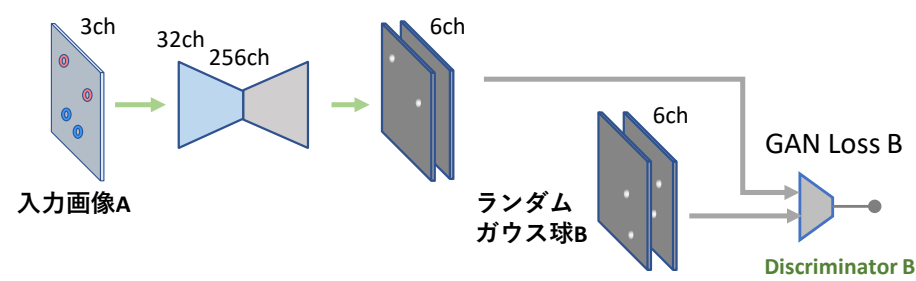
2class検出



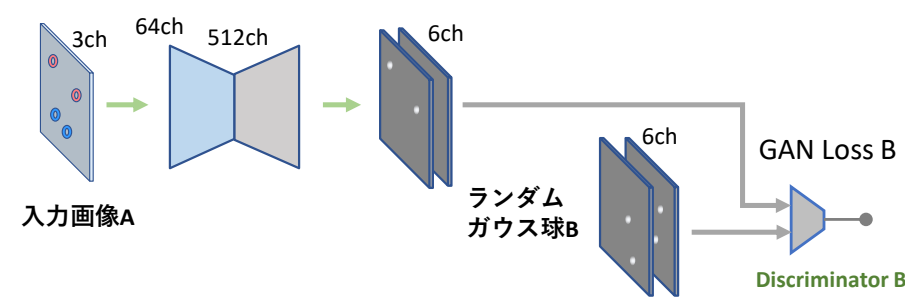
2class検出では2枚分出力させる必要がある

中間チャンネルと分岐の効果

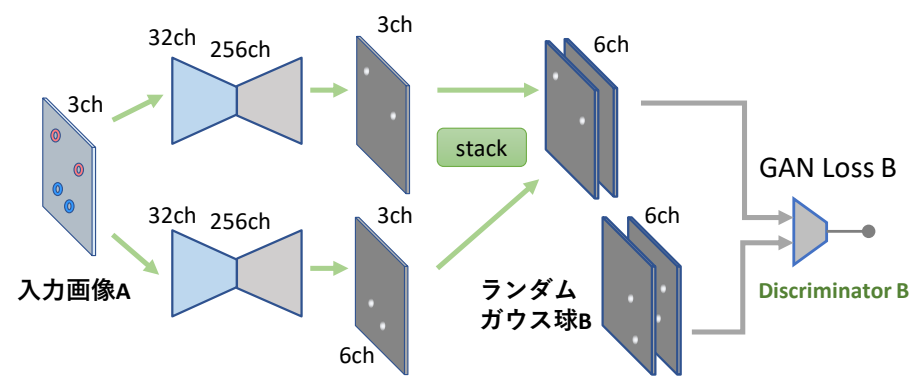
① 1系統 中間ch1倍



② 1系統 中間ch2倍



③ 2分岐 6ch-Dis



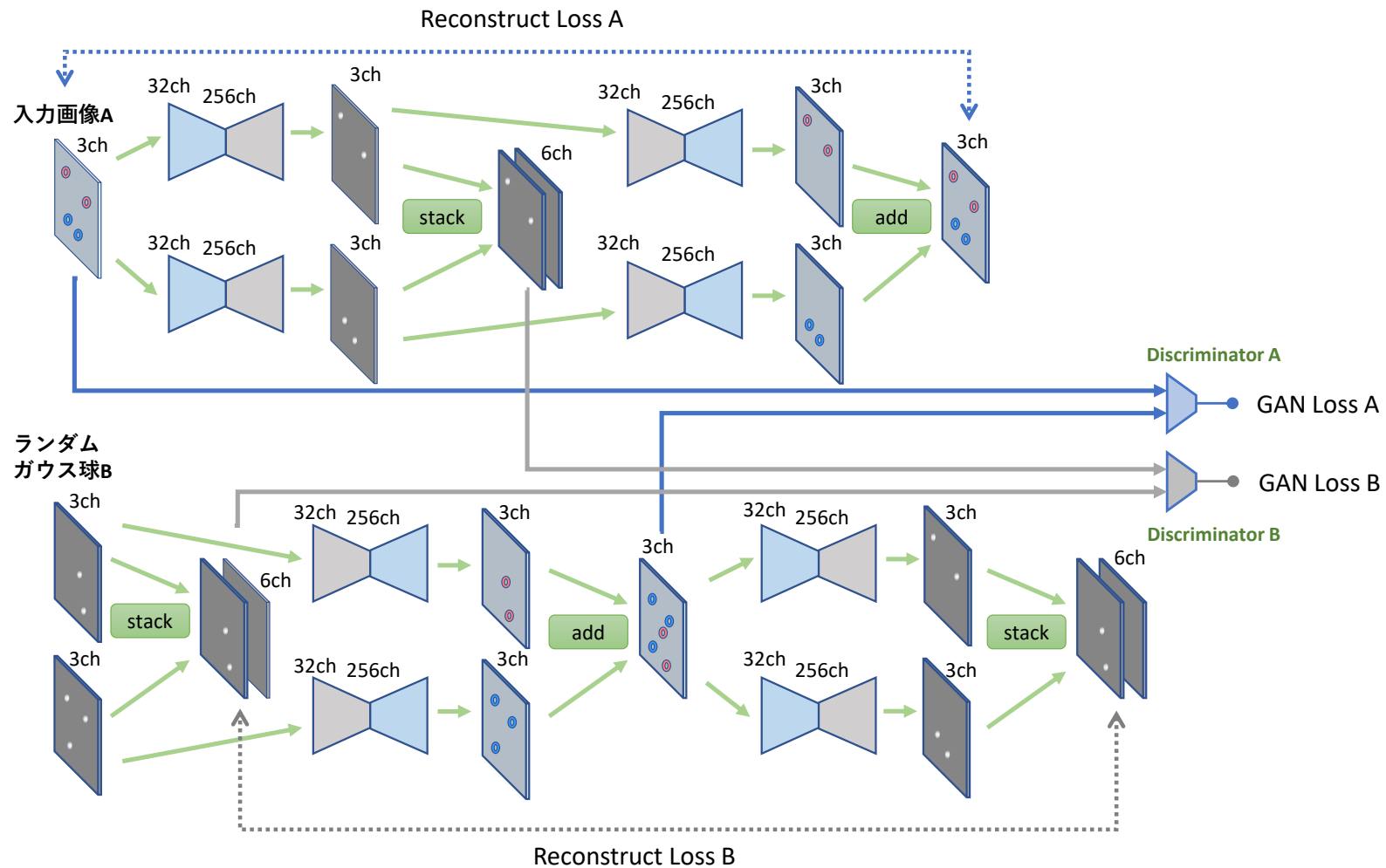
・ 散布数 50/50個

	YG/BG球 [7:3]		YG/BG球 [6:4]	
	平均OK割合	OK※90%以上	平均OK割合	OK90%以上
① 1系統 中間ch1倍	45.9%	15枚	36.8%	3枚
② 1系統 中間ch2倍	71.2%	49枚	40.9%	8枚
③ 2分岐 6ch-Dis	93.5%	91枚	56.3%	32枚

※100回中OKが90%以上の枚数

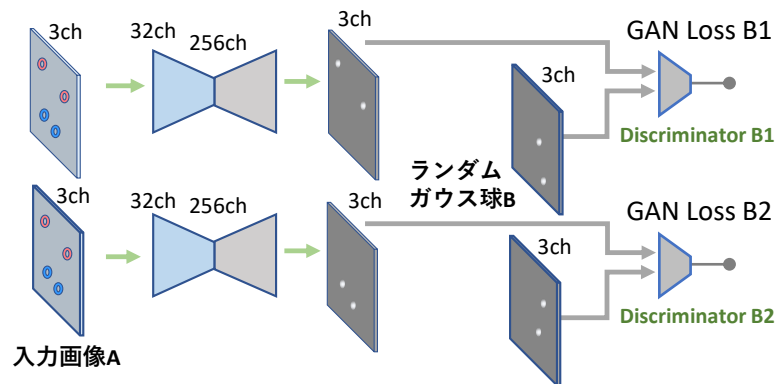
チャンネルを2倍にするよりも、2分岐させた方がよい

2クラス検出 ③ 2分岐 6ch-Dis の全体

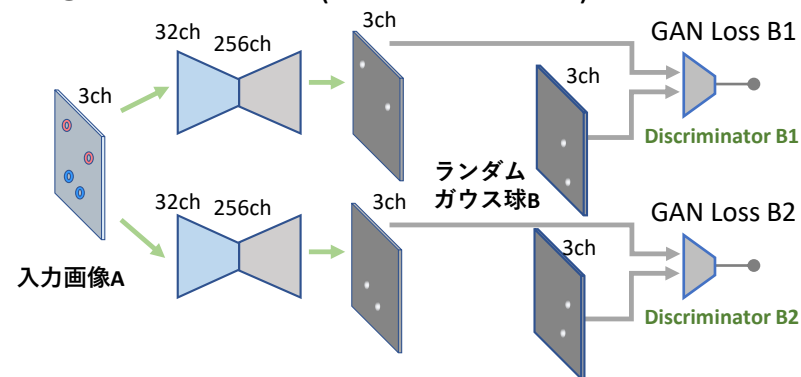


Discriminatorの独立性

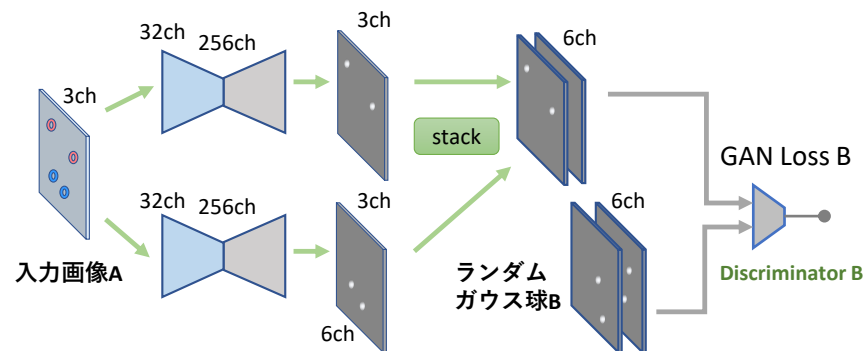
④ 2系統 独立(B⇒Aでaddなし)



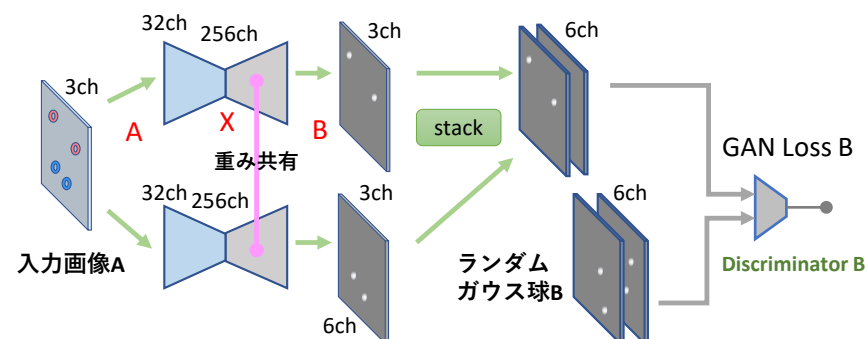
⑤ 2分岐 3ch-Dis (B⇒Aでaddあり)



③ 2分岐 6ch-Dis



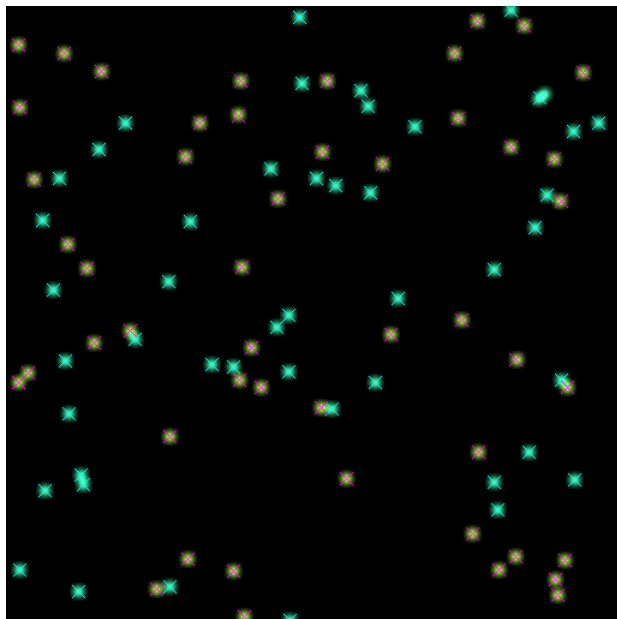
⑥ 2分岐 重み共有



	YG/BG球 [7:3]		YG/BG球 [6:4]		mesh 1pix/2pix		beas 青/赤		nut / washer	
	平均OK 割合	OK 90%以上	平均OK 割合	OK 90%以上	平均OK 割合	OK 90%以上	平均OK 割合	OK 90%以上	平均OK 割合	OK 90%以上
④ 2系統 独立	63.9%	12枚	39.7%	0枚	57.5%	3枚	56.0%	10枚	49.7%	0枚
⑤ 2分岐 3ch-Dis	96.4%	95枚	90.5%	76枚	69.0%	6枚	61.8%	17枚	50.7%	1枚
③ 2分岐 6ch-Dis	93.5%	91枚	56.3%	32枚	73.9%	49枚	57.1%	34枚	63.4%	26枚
⑥ 2分岐 重み共有	93.5%	89枚	57.0%	31枚	70.4%	41枚	48.9%	28枚	54.2%	15枚

3chよりも6chでDiscriminatorをかけた方がよい

YG/BG球 [7:3] 正解 46/49個



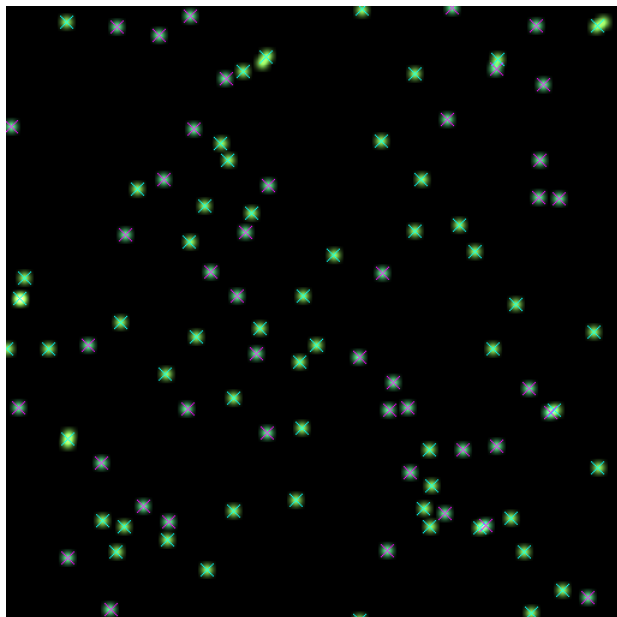
1-3ch出力 46個



4-6ch出力 49個



YG/BG球 [6:4] 正解 59/54個



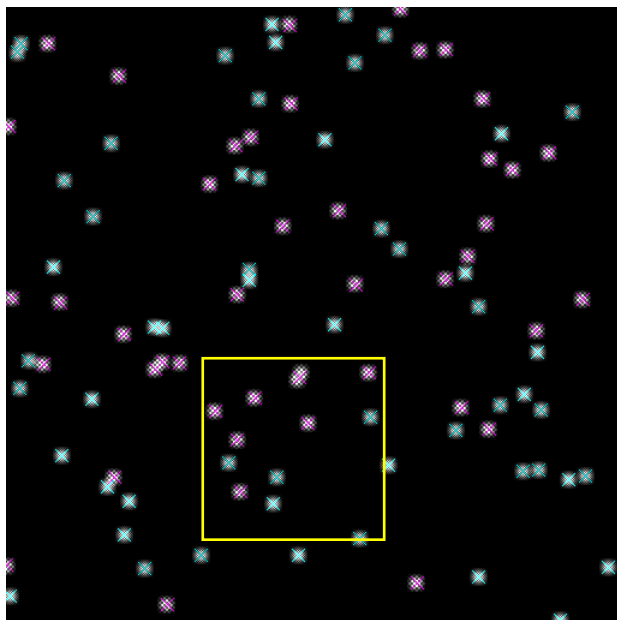
1-3ch出力 55個



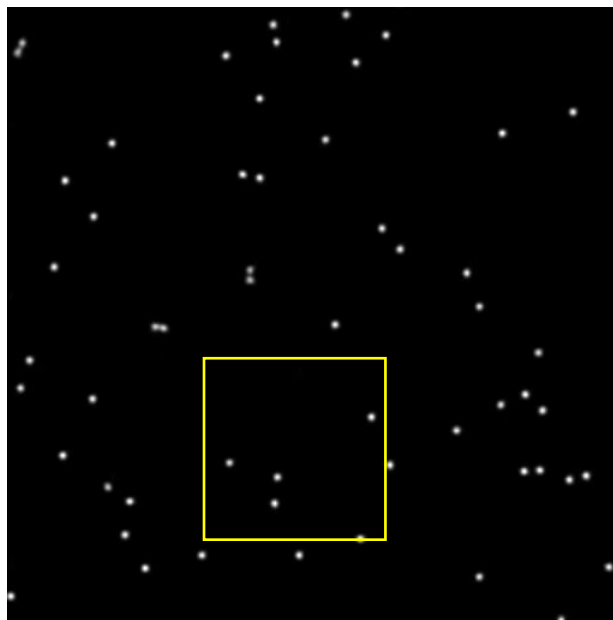
4-6ch出力 54個



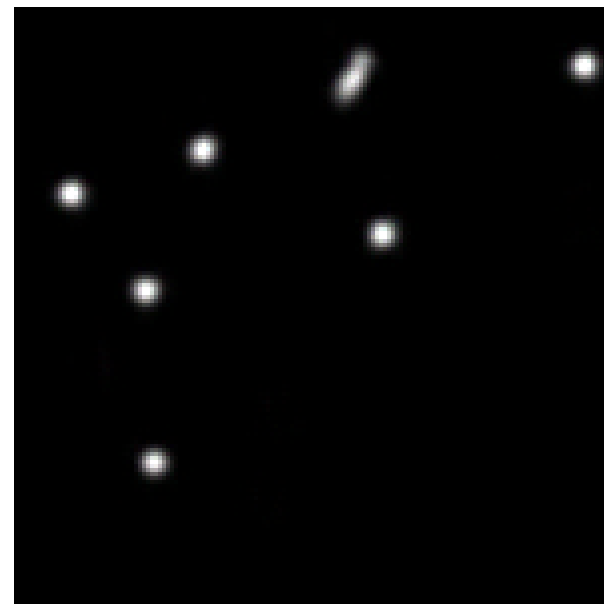
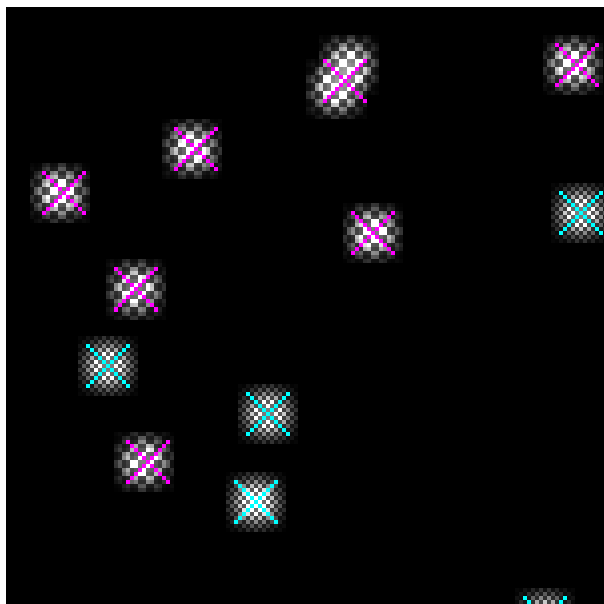
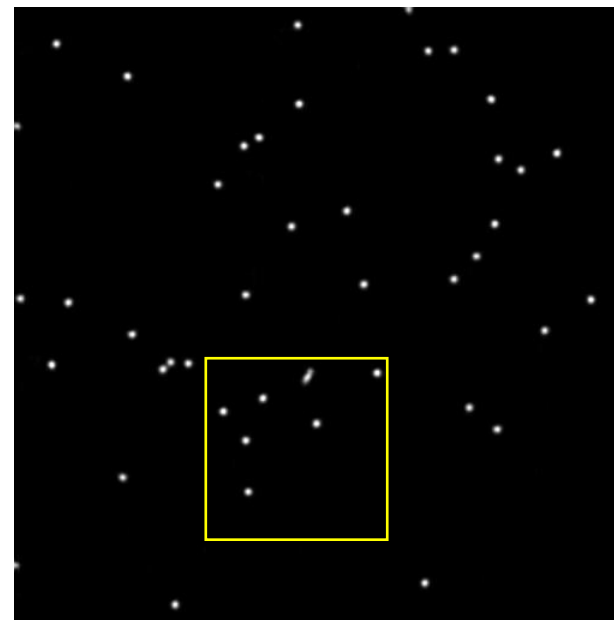
mesh 1pix/2pix 正解56/45個



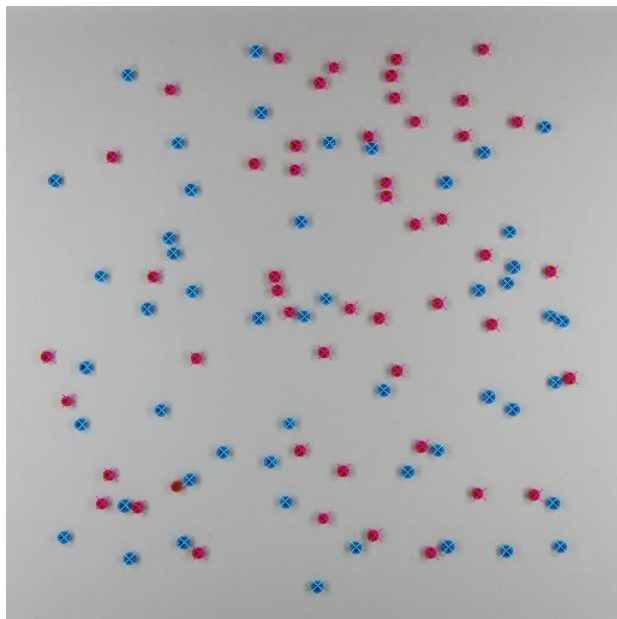
1-3ch出力 56個



4-6ch出力 44個



beads 青/赤 正解50/50個



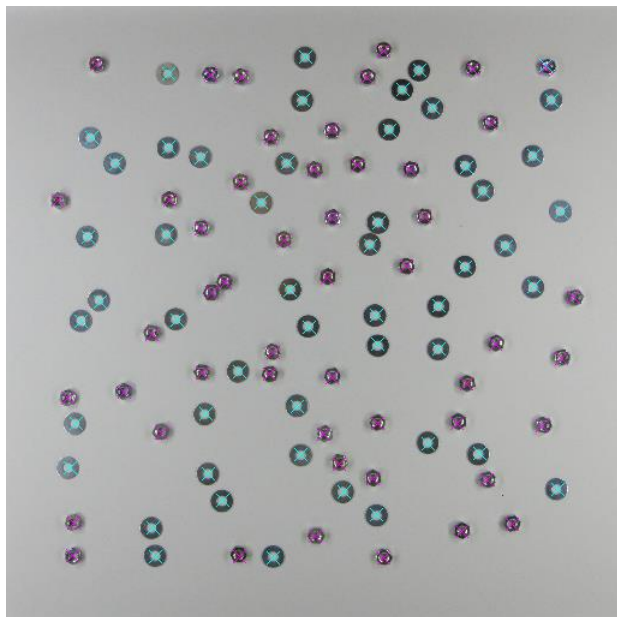
1-3ch出力 50個



4-6ch出力 49個



nut / washer 正解50/50個



1-3ch出力 51個



4-6ch出力 50個



本研究ではDiscoGANを用い、ランダムに散布させたガウス球とのドメイン変換を学習することで、教師なしでの物体検出を試み、1クラスと2クラスの数種のサンプルにおいて物体検出ができた。また、本研究の提案手法の特性について、以下のような知見が得られた。

< 1class検出 >

- ・ 散布数と出力数

⇒ 散布数は正解数の1～2倍の範囲で比較的良好な結果となった

- ・ 教師の有無について比較

⇒ 教師なしは教師ありに近いレベルでモデルが学習できた

- ・ ネットワークのLossの影響

⇒ GAN Bは必須であり、GAN Aは不要となった

< 2class検出 >

- ・ チャンネル数と分岐の効果

⇒ チャンネルを2倍にするよりも、2分岐させた方が良好な結果となった

- ・ Discriminatorのチャンネル数

⇒ 3chよりも6chでDiscriminatorの判定をした方が良好な結果となった

- ・ 2クラス結果の出力例

⇒ 色の違いやパターンの違い、写真において2クラスの物体を分けて検出ができた

補足資料

・ピークポイントの検出方法

例

A

0	0	10	0	0
0	10	30	10	0
10	30	50	30	10
0	10	30	10	0
0	0	10	0	0

B

10	30	30	30	10
30	50	50	50	30
30	50	50	50	30
30	50	50	50	30
10	30	30	30	10

$A*(A==B)$

0	0	0	0	0
0	0	0	0	0
0	0	50	0	0
0	0	0	0	0
0	0	0	0	0

3 × 3 Max Pooling

実際の条件

- ・ 9*9のMax pooling
- ・ 閾値： $255*0.2=51$ 以上のみを抽出