

評価報告書内容(Hazel2)

..ヘーゼルナッツ1の再学習

課題: 以下公開データセットを使って、不良箇所をピクセル単位で判定したい
 データセット: MVTec AD dataset https://www.mvtec.com/fileadmin/Redaktion/mvtec.com/company/research/datasets/mvtec_ad.pdf
 サブセット: ヘーゼルナッツ
 不良品クラス: 2クラス (1: Crackカケ, 2: Print印字) Crackにはヘーゼルナッツ1におけるCut, Holeを統合
 解析タスク: セマンテック セグメンテーション

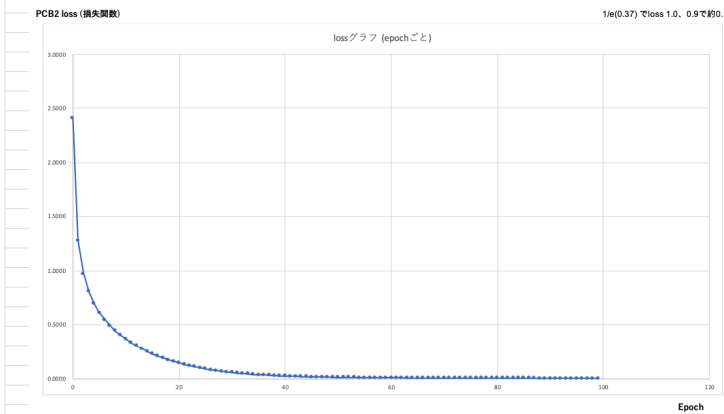
学習データ	総数	うち学習用	うち検証用	推論テスト	*単位: 画像枚数
良品数	431	128	58	0	良品、不良品のデータ不均衡を考慮し、バランシングのためランダムサンプリングを実施 *ランダムサンプリング実施データ *不良品数が少ないため検証用と同一データを使用
不良品数	40	58	12	12	
合計	471	186	70	12	

画像サイズ 1404 x 1070 **2.2 : 1 の割合** **4.8 : 1 の割合**
 ラベリング グレースケール画像 (クラスごとに白黒画像だったので、上記不良クラスのIDIに変更 0が不良品でない背景部分)

入力画像サイズ 512 x 512 (RGB)。ストライド384 (1/4である128pixel分を再利用するため)
 前処理 png -> jpg変換。maskの値を255 -> classidに変更。1024 x 1024サイズはそのまま利用。
 パラメータ調整 Hazel1に対し、Printが相対的に少なくなったため、Lossウェイトを増やした。あとOptimizerをPCB2と同様Adadeltaに。
 最終Epoch数 100 (1 Epoch 約6分)

予測モデル評価 *訓練された学習モデルの事
 使用データ 計70 (うち不良品12)
 採用モデルEpoch数 100 lossで判定
 lossグラフ

*学習の進み具合を知る簡易的な予測モデル性能の評価指標
 過学習しているかどうかについてはこの数字では判断できない



2クラス評価 (良品/不良品)

混合行列		予測		*単位は画素数
		Defect (Positive)	BG (Negative)	
GT	Defect	146968	163967	FN: 見逃し数 FP: お手付き数
	BG	586070	72503315	

*混合行列 (Confusion Matrix)
 予測モデルの性能評価指標として一般的に用いられる。
 縦軸視点が予測の内訳。横軸が教師データ内訳となる。
 *GT: 教師データを指す(Ground Truth)

*1画像から6枚分の512x512データ作成。計280画像分のpixel数内訳となる。

IoU	0.1638	TP	TP+FP+FN
		146968	897005

*IoU: Intersection over Union (Jaccard係数とも)
 評価式: $IoU = TP / (TP + FP + FN)$
 本課題データは良品(背景)領域が圧倒的に多いためTNを使用しない評価指標として採用。
 TP: 不良品領域だと感知したら正解だったケース (言い当て)(True Positive: 真陽性)
 FP: 不良品領域だと感知したが不正解だったケース (お手付き)(False Positive: 偽陽性)
 TN: 良品領域と考え感知しなかったケース (感知せず正解)(True Negative: 真陰性)
 FN: 良品領域と考えたが不良品だったケース (見逃し)(False Negative: 偽陰性)

3クラス評価(不良品クラス毎)

混合行列		予測		
		P-Crack	P-Print	N-BG
GT	Crack	66685	1	147329
	Print	70821	9461	16638
	BG	585907	163	72503315

IoU	0.0849	TP	TP+FP+FN
		76146	897005

予測テスト

平均推論速度 (推論後の画像解析(領域化等)の時間は含まない)
 CPU: 調査せず
 GPU: 0.0425秒/1画像 (約24FPS) ...GeForce RTX 2080 Ti

上手く検出できた画像1

*特筆すべきデータをピックアップした

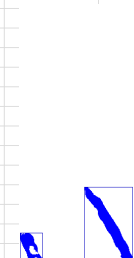
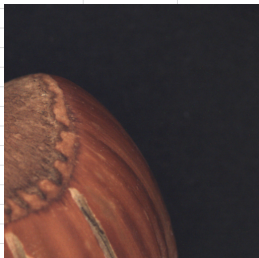
オリジナル画像

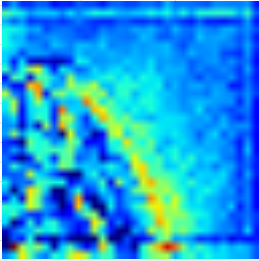
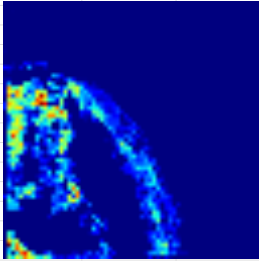
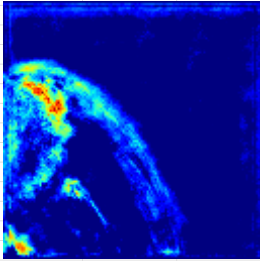
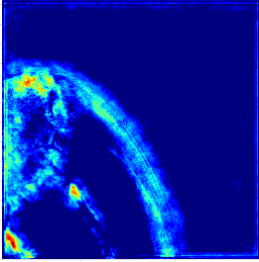
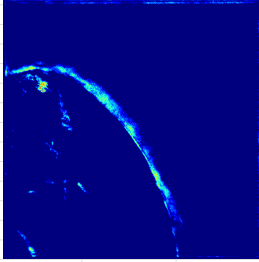
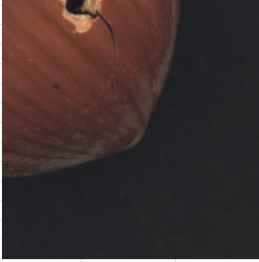

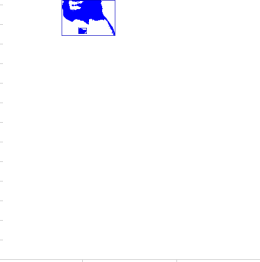
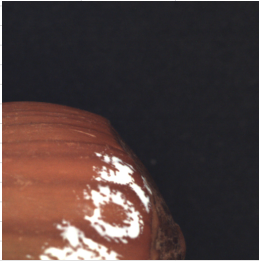

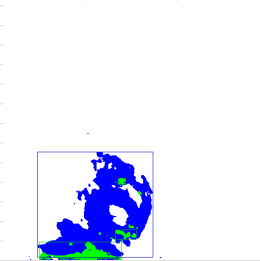

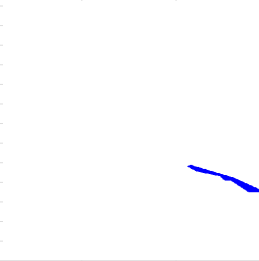

正解画像

推論画像

白: 背景
 青: Crack (Cut, Hole含む)
 黄緑: Print

*pixel領域100以上に対して矩形表示



ヒートマップ			Grad-CAMを使用。予測モデルが注目している領域ほどより赤(暖色)で反応
層1 (解像度32 x 32)	層2 (解像度64 x 64)	層3 (解像度128 x 128)	
			
層4 (解像度256 x 256)	層5 (解像度512 x 512)		
			
上手く検出できた画像2 オリジナル画像	正解画像	推論画像	ヒートマップは割愛
			
上手く検出できた画像3 オリジナル画像	正解画像	推論画像	不良品クラスの偏り(Crack - Print = 3 : 1)のため Printが検出されにくい状況ではあった
			
上手く検出できなかった画像 オリジナル画像	正解画像	推論画像	ごちゃごちゃしている中の傷の検出は 全体的に弱かった
			
所見	<p>Crack、切れ目、穴は様相が似ているため、データを統合したが、そのため、単純なCrackの検出はほぼほぼできた。ただ、上記の例の様に ごちゃごちゃ感のある中に存在するCrack検出は全体的に弱かった。またCrackクラスを統合する前はPrintの検出が上手くいっていたが、 今回検出されにくかったのは、不均衡データになってしまったからだと考えられ、多少のloss関数へのウェイトを適用したがあまり効果が なかったのかも知れない。</p> <p>いずれにしてもPCB2と比べ、データ数や不均衡といった不利な部分もあったが、これだけ見た目にも分かりやすいので、データ数を増やしたり また、画像全体が暗い中での撮影のため、露出を上げたりすれば、もっと意図通りの検出ができると思われる。</p> <p>以上でPoC検証作業を完了とする。</p>		