



深層学習を用いた
SAR 画像からの船舶識別
及びFPGAの実装に関する一検討

Shintaro Hashimoto,
Yohei Sugimoto, Ko Hamamoto and Naoki Ishihama, JAXA

2018/2/16

本日の内容



深層学習を用いてSAR画像から船舶を識別した
研究結果とFPGA化についてご報告致します。

① Introduction

研究背景、目的、研究手順

■ 安全保障の観点

- ・ 特定海域における外国漁船などの不審船の活動が活発化してきている
 - 国内の海洋に係る機関はEEZ周辺域を把握したい
- ・ 安全保障に影響を及ぼす違法漁業や海賊などの不審船は一般的に船舶情報(AIS)を発しておらず、入港時・航行時に監視・管理ができていない
 - AISに頼らない何らかの客観視可能な手法で船舶を識別する必要がある
 - EEZ周辺域を人が全て監視することは難しいため自動で検出する必要がある

■ 衛星側の観点

- ・ 地上間通信の帯域的制約により、衛星により撮影した全データを地上にダウンロードできていない（現状、データを取得していない）
 - 衛星側で有用な情報かどうかを判別して必要なデータのみを保存させたい

■ 安全保障の観点

- ・ 特定海域における外国漁船などの不審船の活動が**活発化**してきている

→ 国内の海洋に係る機関はEEZ周辺域を把握したい

尖閣諸島周辺海域において退去警告隻数は、

8隻 → 208隻^[1]
(平成23年) (平成26年)

[1] 海上保安庁, 海上保安体制強化に関する方針について (2016)

■ 衛星側の観点

- ・ 地上間通信の帯域的制約により、衛星により撮影した全データを地上にダウンロードできていない（現状、データを取得していない）

→ 衛星側で有用な情報かどうかを判別して必要なデータのみを保存させたい

■ 安全保障の観点

- ・ 特定海域における外国漁船などの不審船の活動が活発化してきている
→ 国内の海洋に係る機関はEEZ周辺域を把握したい
- ・ 安全保障に影響を及ぼす違法漁業や海賊などの不審船は一般的に船舶情報(AIS)を発しておらず、入港時・航行時に監視・管理ができていない
→ AISに頼らずとも衛星側で可能な手段で船舶を識別する必要がある

船舶自動識別装置 (AIS: Automatic Identification System) :

AISは大型船舶などの特定船舶に設置が義務付けられている。AISは電波を用い航行情報(識別符号、船名、位置、針路、速力、目的地など)を相互に交換することで洋上を航行する船舶同士の安全航行を支援する。

[2] International Maritime Organization, "SOLAS", Chapter V, 2002.

- 衛星側で有用な情報かどうかを判別して必要なデータのみを保存させたい

■ 研究目的

SAR画像を入力とし、船舶の位置、長さ、種類を識別し、出力するシステムの構築。これによって不審船の発見及び早期処置に資する

■ 研究手順(ロードマップ)

- 1) SARのL1.5 Dataから船舶の存在を識別する (地上)
- 2) SARのL1.5 Dataから船舶の存在や種類などを識別する (地上)
- 3) 幾つかのアルゴリズムを検討し識別精度の向上を目指す (地上)
- 4) 開発したアルゴリズムをFPGAに搭載する (地上)

----- 将来的 -----

- 5) 衛星に搭載する (宇宙)



② Method

アプローチ、関連研究、研究手法

方針及びSAR L1.5 の海上シーンの特徴

■ 方針

SAR L1.5 Dataを画像として扱う。この画像から船舶の特徴量を取得し、船舶識別を行う。

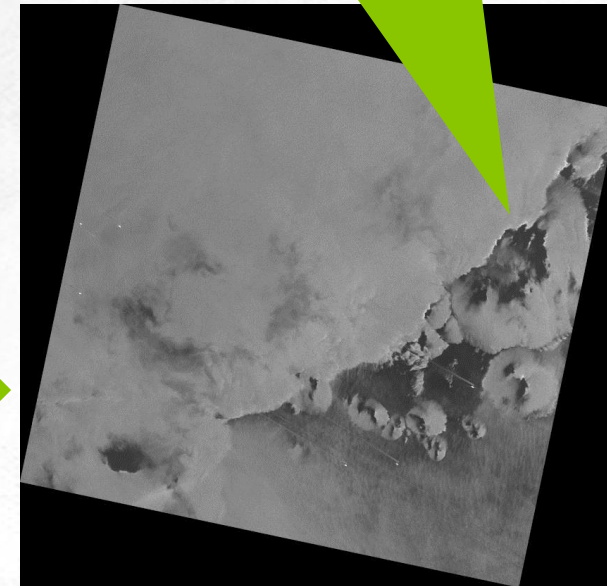
ALOS-2のLバンドSAR(PLASAR-2)のHV偏波^[3] 10m分解能

■ SAR L1.5の海上シーンの特徴

- ・ グレースケール画像かつ輝度が全体的に低い
- ・ 画像解像度が高い (13, 000 px × 13, 000px程度)
- ・ 船舶の占める特徴量が小さい (40 px × 40px 程度)
- ・ ノイズ(海面クラッタ、対流、サイドローブ)が強い
- ・ 船舶間の距離が近い
- ・ 船舶の特徴が極めて分かりづらい



この模様は対流及び波
だと考えられる



調整画像

[3] 木村 広和, 吉野辰萌 (株式会社イメージワン), “RADARSAT-2 衛星のSAR画像を用いた判読検証”, 日本写真測量学会 平成20年, 2008.

方針及びSAR L1.5 の海上シーンの特徴

■ 方針

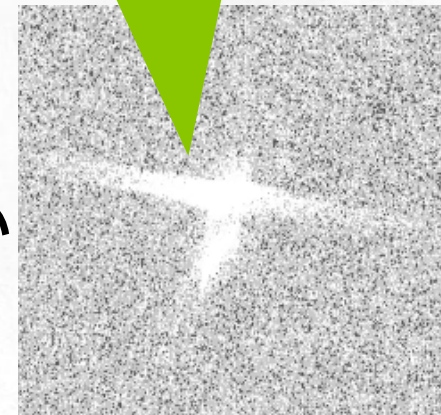
SAR L1.5 Dataを画像として扱う。この画像から船舶の特徴量を取得し、船舶識別を行う。

■ SAR L1.5の海上シーンの特徴

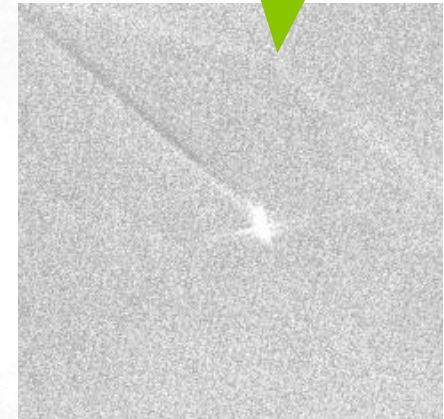
- ・ グレースケール画像かつ輝度が全体的に低い
- ・ 画像解像度が高い (13, 000 px × 13, 000px程度)
- ・ 船舶の占める特徴量が小さい (40 px × 40px 程度)
- ・ ノイズ(海面クラッタ、対流、サイドローブ)が強い
- ・ 船舶間の距離が近い
- ・ 船舶の特徴が極めて分かりづらい

サイドローブの影響が強くて
どちらが船首か分からない

航跡やノイズ(波？対
流？)が現れている



船舶画像 ①



船舶画像 ②

■ 特徴量抽出のために考えられた手法

- ・ 閾値フィルタリング手法
 - CFAR^[4]、標準偏差フィルタ^[5]
- ・ 古典的機械学習手法
 - Haar-Cascades (AdaBoost)^[6]
- ・ 深層学習による手法
 - Deep learning (CNN)^[7]

[4] Jiaqiu Ai, et al., "A CorrelationBased Joint CFAR Detector Using AdaptivelyTruncated Statistics in SAR Imagery", Sensors (Basel), 2017.

[5] Motofumi Arie, "Improvement of ship-sea clutter ratio of SAR imagery using standarddeviation filter ", Geoscience and Remote Sensing Symposium (IGARSS), 2011 IEEE International, 2011.

[6] Pual Viola, Michael Jones, Rapid Object Detection using a Boosted Cascade of Simple Features, CVPR, 2001.

[7] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, 2012.

■ 閾値フィルタリング手法

- ・ CFAR

- 処理速度が高速。ノイズから船舶(ピーク)を抽出でき、精度は高い。

- ・ 標準偏差フィルタ

- 処理速度が高速。特に海面クラッタなどのノイズに強く、抽出精度は高い。

■ 古典的機械学習手法

- ・ Haar-Cascades

- 処理速度は中。明暗の2値化情報から識別するため、ノイズと船舶の識別が困難。

■ 深層学習による手法

- ・ Deep learning (CNN)

- 処理速度は遅。形に加えて輝度値も学習させることができるため、複雑な識別が可能。

船舶識別手法の特徴一覧



■ 閾値フィルタリング手法

・ CFAR

- 処理速度が高速。ノイズから船舶(ピーク)を抽出でき、精度は高い。

・ 標準偏差フィルタ

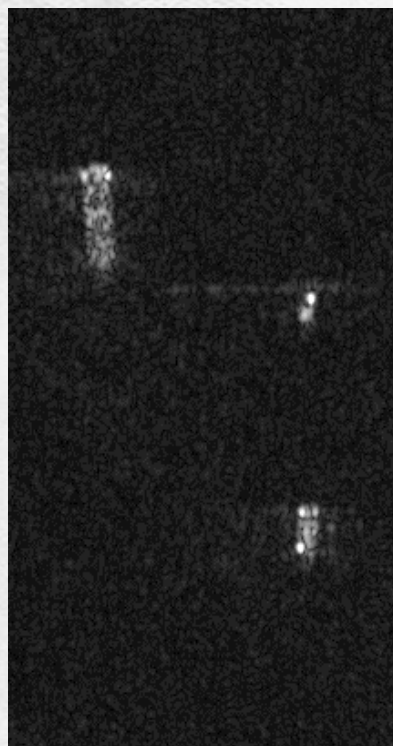
- 処理速度が高速。特に海面クラッタなどのノイズに強く、抽出精度は高い。

■ 古典的機械学習手法

高速で船舶の抽出をえることは魅力的である。しかし、2値化(フィルタリング)アルゴリズムのため、船長や船種の識別まではできない。また海上シーンの非一様性に応じて閾値を人間が設定する必要がある。

■ CFAR (Cell Averaging CFAR)

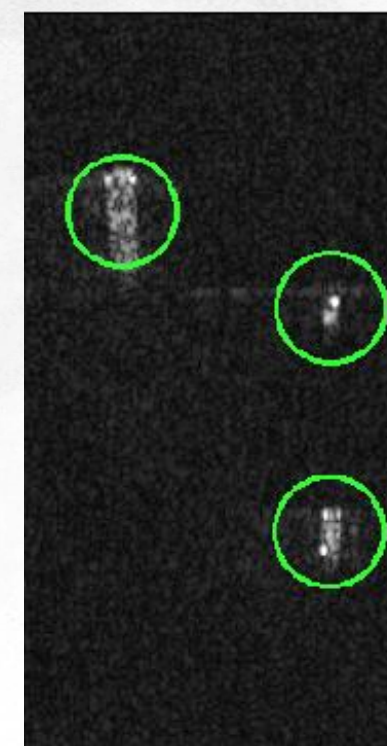
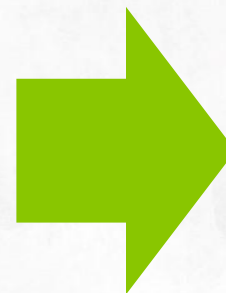
- ・ 検出精度は殆ど100%に近く、精度が高い



元画像 (HV)*



CFAR



船舶抽出

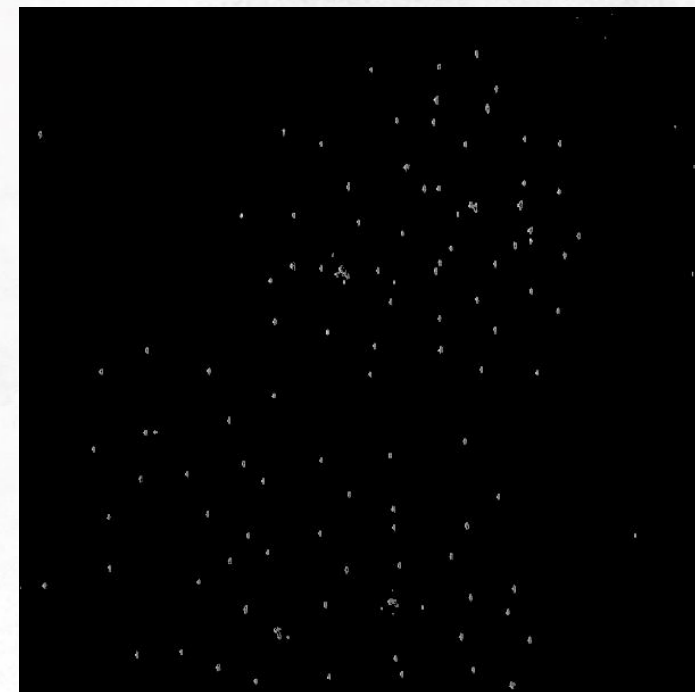
* JAXAのシステム技術ユニット所属、杉本様からの提供

■標準偏差フィルタ

- ・ 32隻 (3シーン)識別したところ100% (処理時間は平均30秒)



元画像 (HV)*



標準偏差フィルタ

* JAXAのシステム技術ユニット所属、杉本様からの提供

■標準偏差フィルタ

- ・ 32隻 (3シーン)識別したところ100% (処理時間は平均30秒)



【注意】

CFARも標準偏差フィルタも信号(輝度値)の強度を見ているだけで、船舶については見ていない。つまりは信号のピーク(特徴)を抽出しているだけで、その形は識別できていない。



元画像 (HV)*



標準偏差フィルタ

* JAXAのシステム技術ユニット所属、杉本様からの提供

明暗(2値化)情報からのみだと特徴量の少ない船舶とノイズを精緻に識別することは困難。また、船舶の識別ができたとしても、船舶解像度が低いため船長及び船種までの識別はできない。

- 処理速度が高速。特に海面クラッタなどのノイズに強く、抽出精度は高い。

■ 古典的機械学習手法

・ Haar-Cascades(Adaboost)

- 処理速度は中。明暗の2値化情報から識別するため、ノイズと船舶の識別が困難。

■ 深層学習による手法

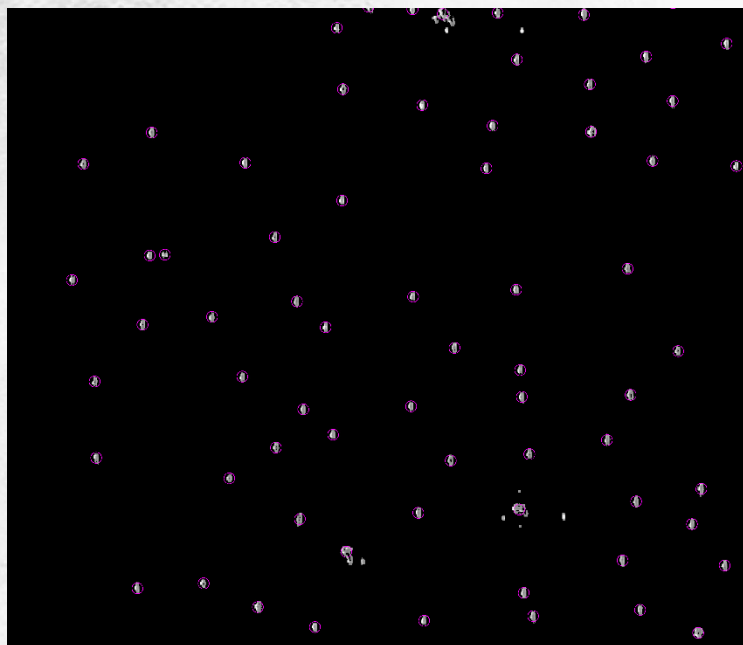
・ Deep learning (CNN)

- 処理速度は遅。形に加えて輝度値も学習させることができるため、複雑な識別が可能。

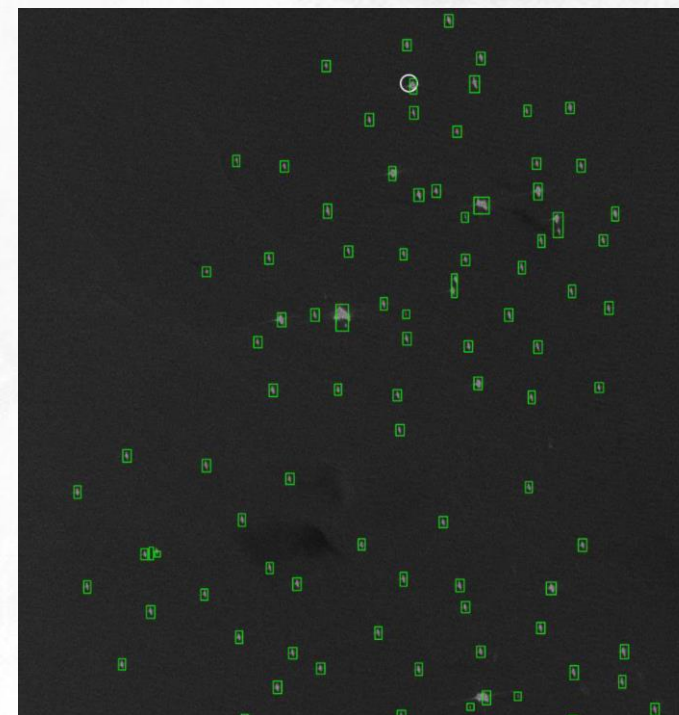
Haar-Cascades vs CNN

■ Haar-Cascades vs CNN

- ・ Haar-Cascadesによる検出精度は32隻 (3シーン)識別したところ59% (処理時間は平均8秒)
- ・ CNNによる検出精度は同条件下で100% (処理時間は平均3,000秒)



Haar-cascadesによる検出 *



CNNによる検出

* JAXAのシステム技術ユニット所属、杉本様からの提供

船舶識別手法の特徴一覧



■ 閾値フィルタリング手法

・ CEAP

畳み込みニューラルネットワークのため、人間がルールを与える必要がない。どのような輝度値のパターンがどの船舶(性質)を表しているかを勝手にルール化してくれる。このため、船舶のあらゆる識別が可能。

- 処理速度は中。学習済み情報から識別するため、ノイズと船舶の識別が困難。

■ 深層学習による手法

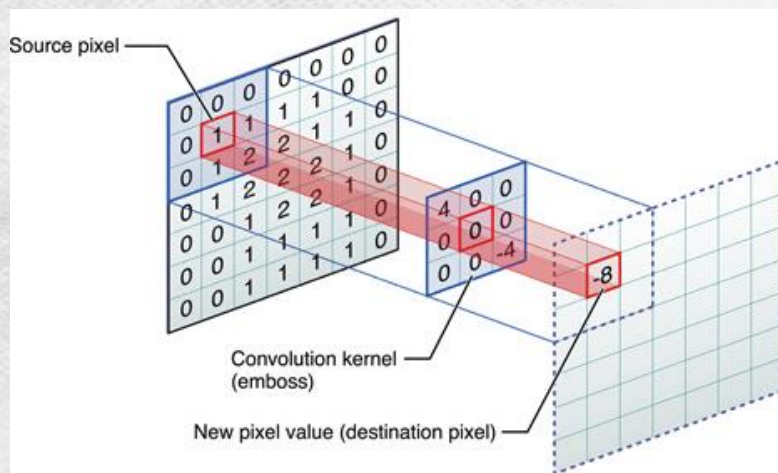
・ Deep learning (CNN)

- 処理速度は遅。形に加えて輝度値も学習させることができるため、複雑な識別が可能。

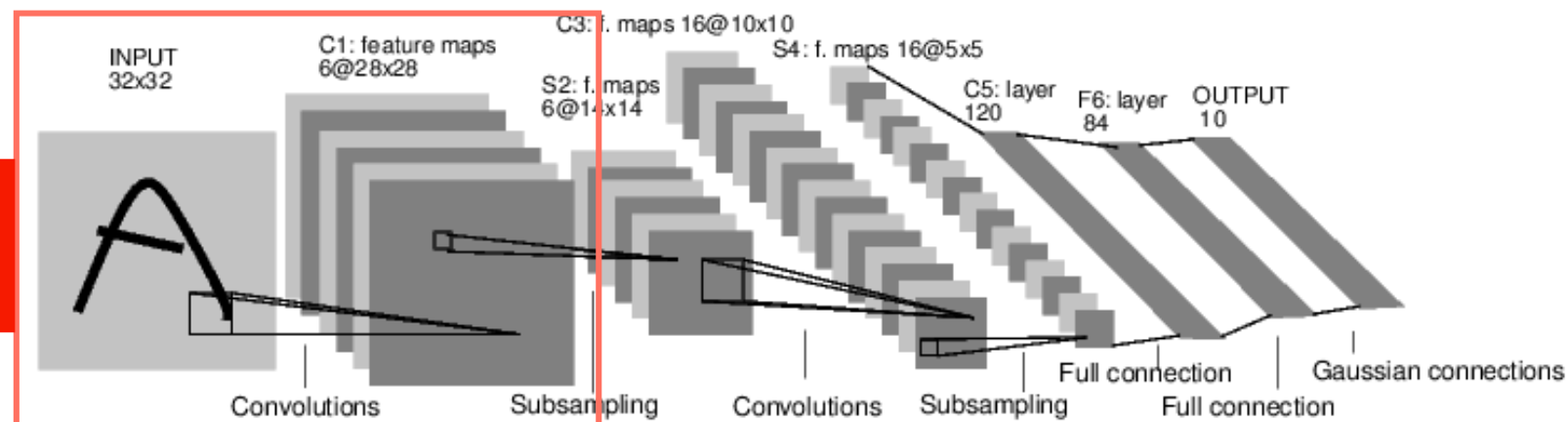
Convolution Neural Network (深層学習)



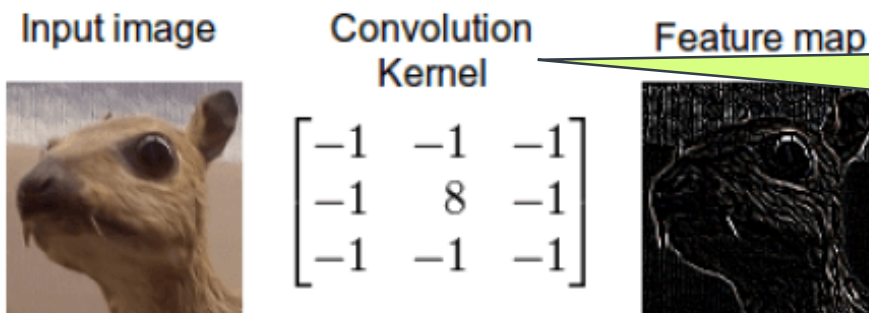
■ Convolution層



■ CNNの全体像



[8] Yann LeCunn, Leon Bottou, Tyoshua Bengio and Patrick Haffner, "Gradient-Based Learning Applied to Document Recognition", IEEE, 1998.



■ 従来の機械学習と比べて

- ・フィルタを学習によって獲得する
- ・フィルタの畳み込みを多層で行う

■ 閾値フィルタリング手法

・ CEAP

【採用】

船長、船種が識別可能な上に、Haar-Cascadesのような古典的機械学習の手法よりも精度が高いことが確認できたため、こちらの手法を採用。

- 処理速度は中。形状・輝度情報から識別するため、ノイズと船舶の識別が困難。

■ 深層学習による手法（採用）

・ Deep learning (CNN)

- 処理速度は遅。形に加えて輝度値も学習させることができるため、複雑な識別が可能。

CNNを用いた船舶識別の関連研究



■ ESAの船舶識別の研究^[9]

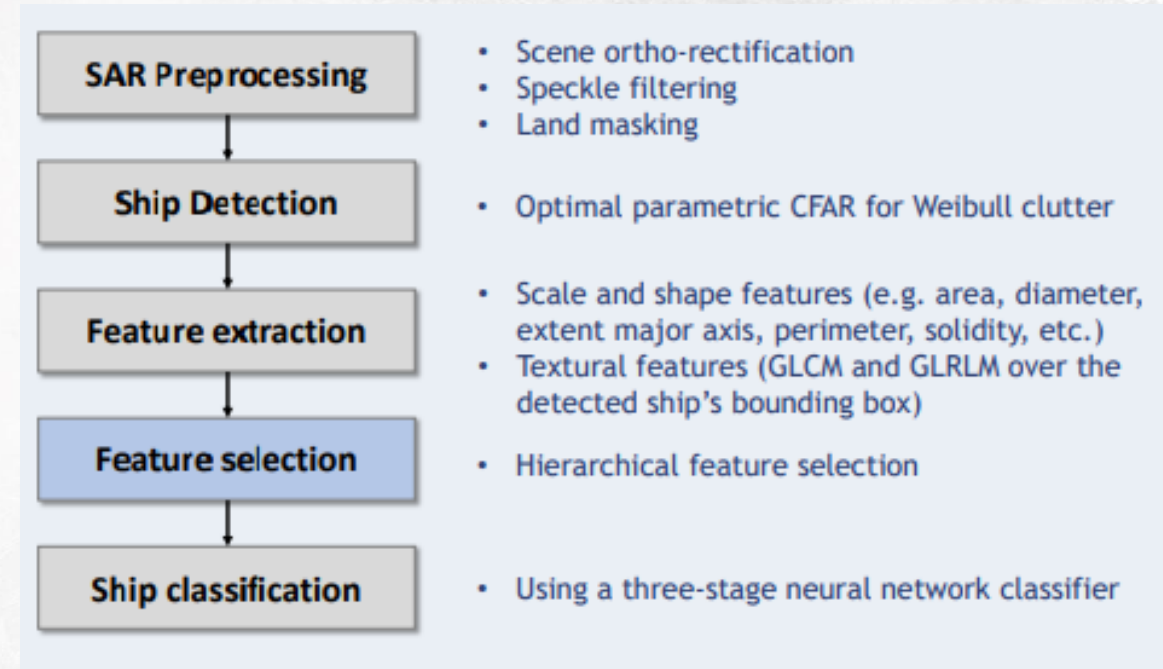
- 船舶識別の前に前処理としてフィルタリングを掛けて各船舶を抽出している

→ 前処理もCNNによって行うことで航跡なども船舶の情報として扱えるようになる

- 階層的に船舶識別を行っている

→ 船種識別に留まっており、船長の識別までを行っていない

* なお、とある製品では船長識別が可能であるが標準偏差フィルタによって抽出された船舶に対して行うため、サイドローブの影響などにより2倍の誤差がある

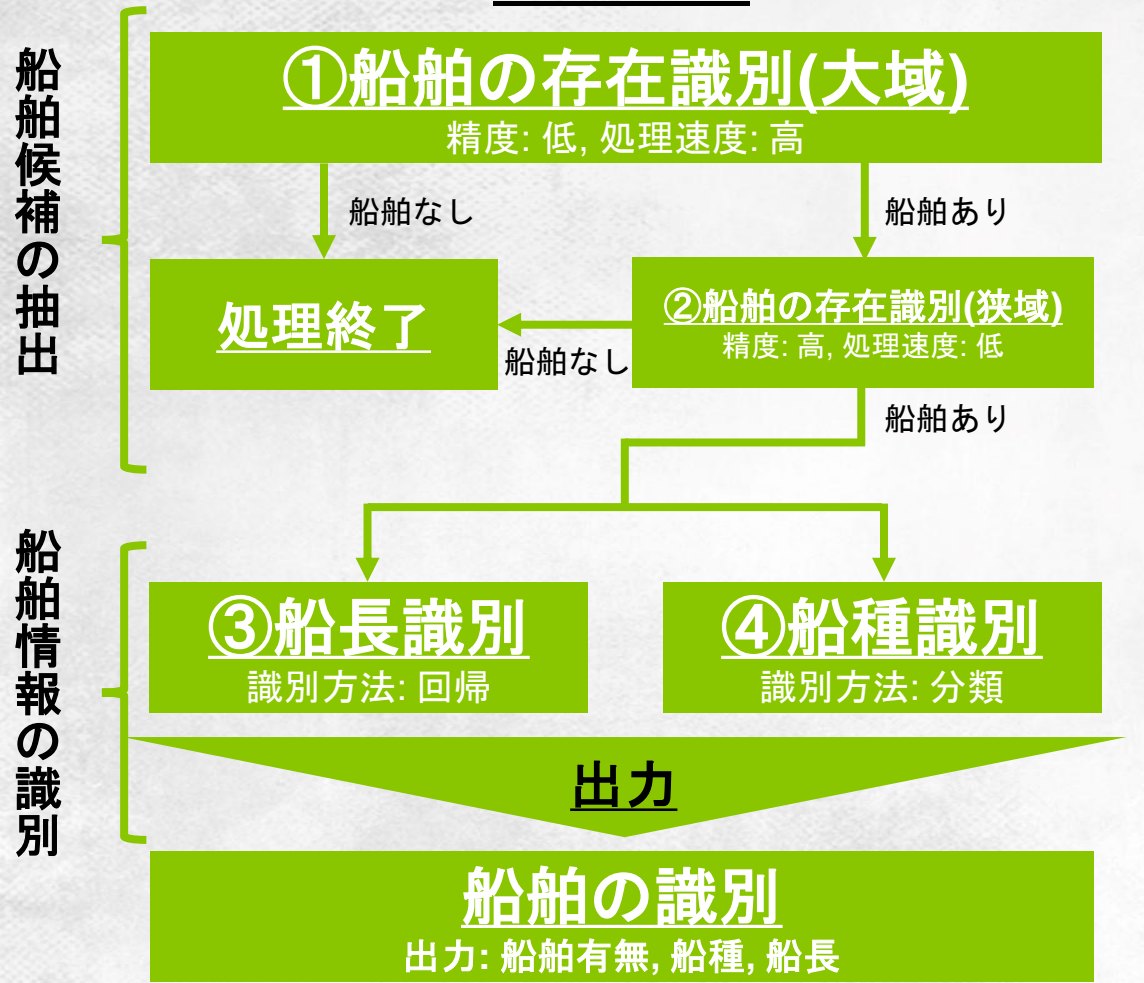


[9] A.Makedonas, C. Theoharatos, V.Tsagaris, S.Costicoglou, “A MULTILEVEL APPROACH TO SHIP CLASSIFICATION ON SENTINEL-1 SAR IMAGES USING ARTIFICIAL NEURAL NETWORKS” , LPS16 ESA Symposium, 2016.

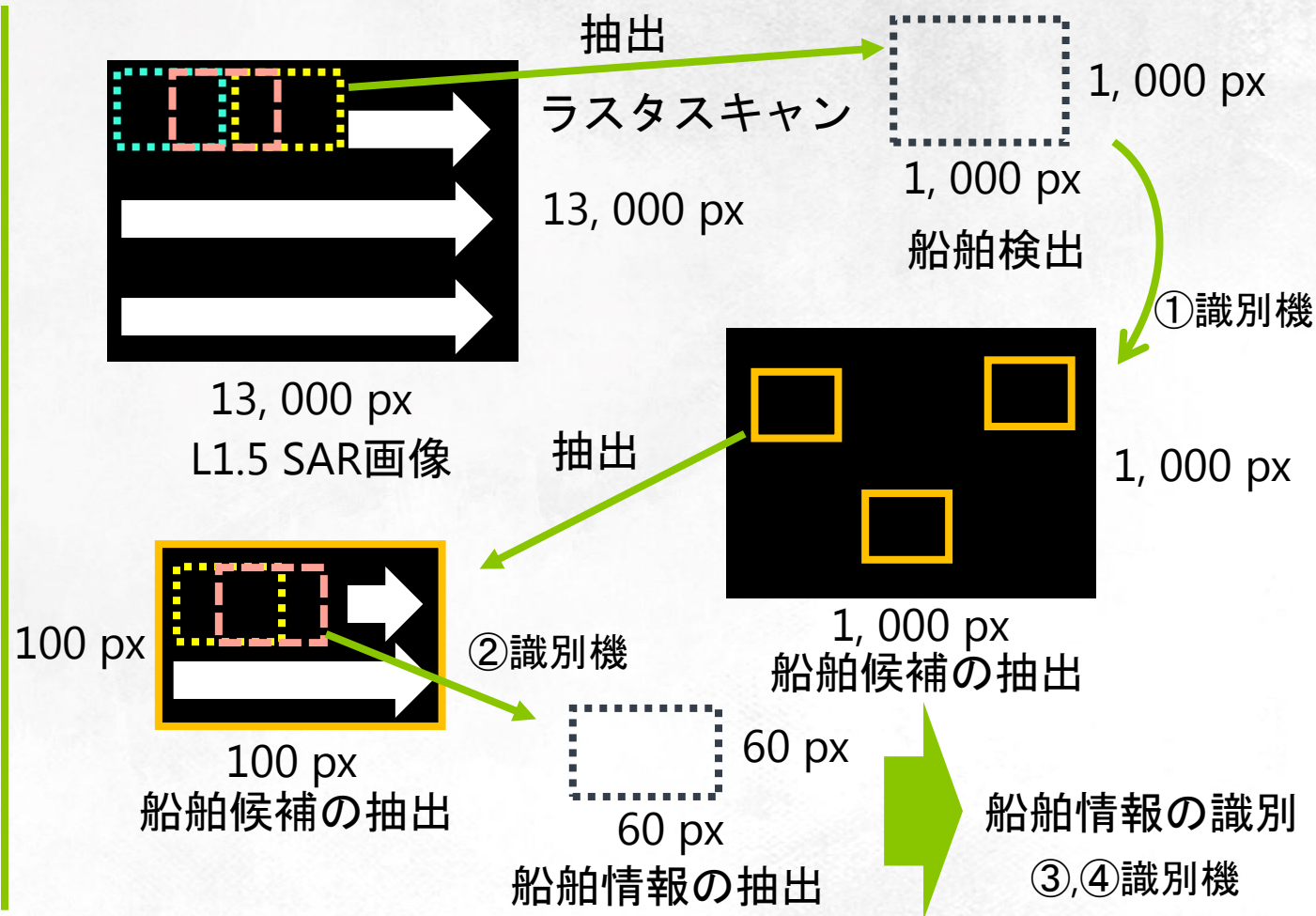
本研究における 船舶識別のアプローチ

船舶識別のアプローチ

全体像

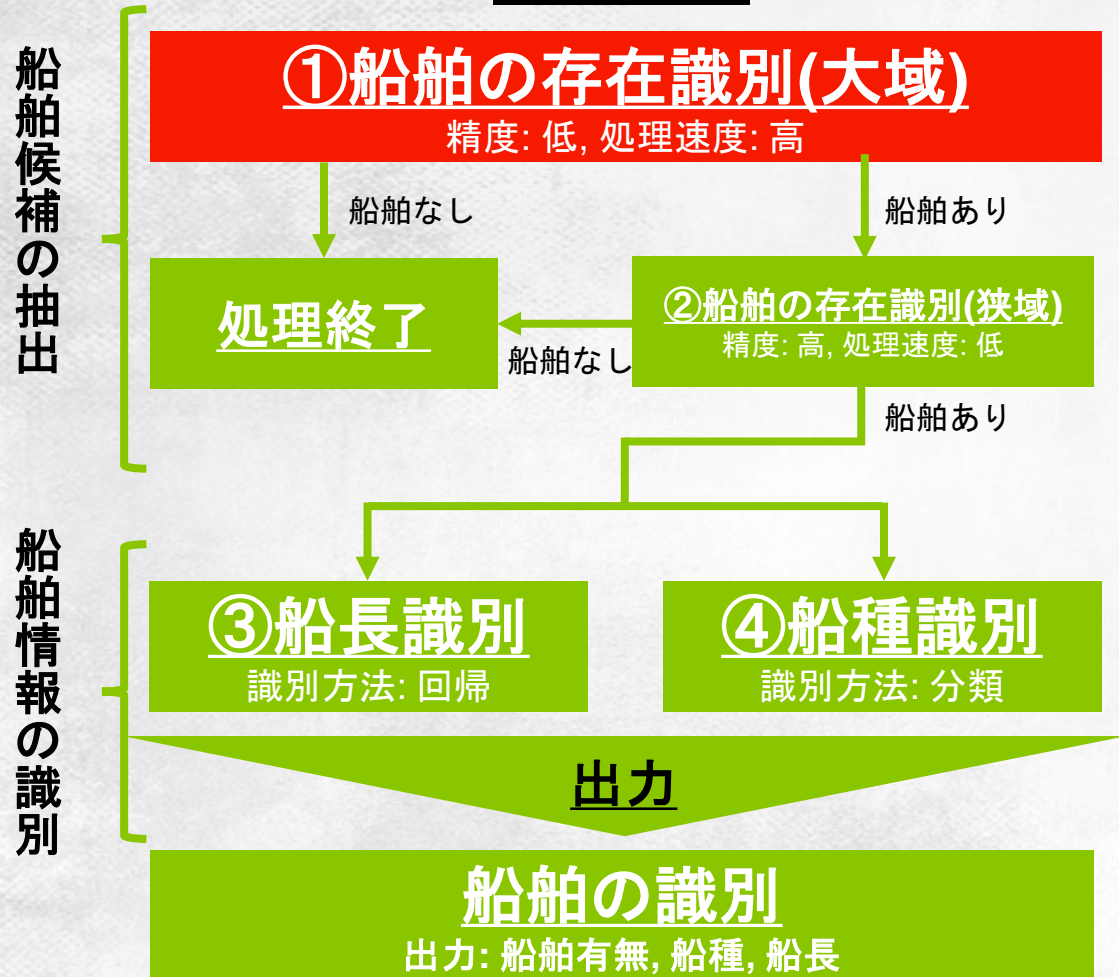


全体像の具体例



船舶識別のアプローチ

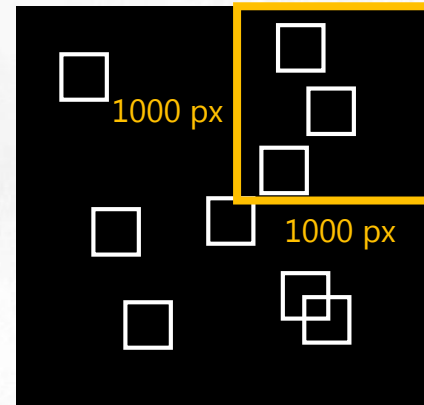
全体像



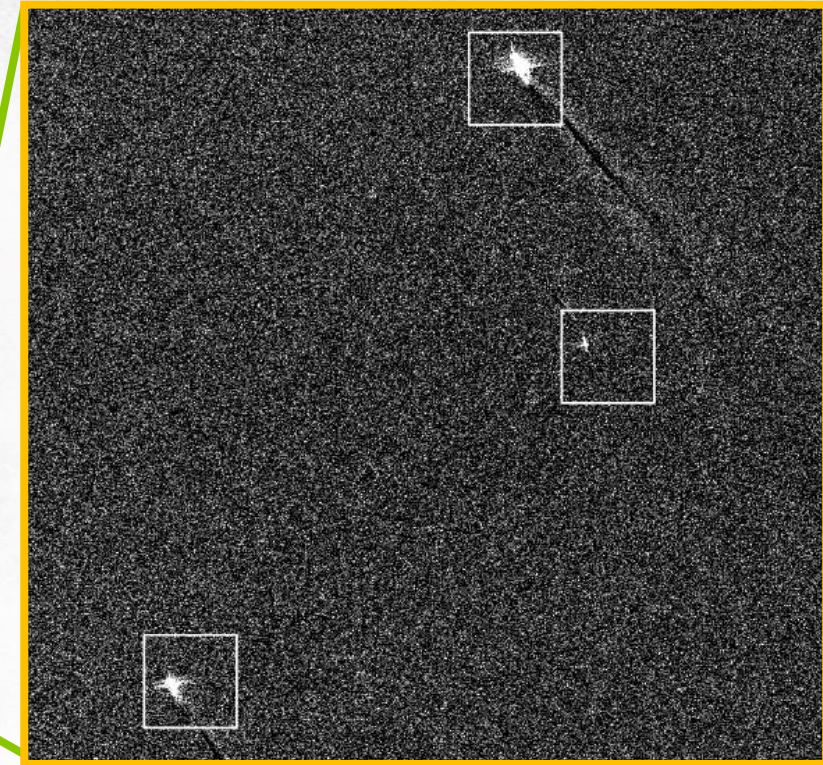
大域画像から船舶の候補を抽出

船舶の候補を抽出

- ・ 1シーン7秒以内
- ・ 感度変更可能
- ・ 検出画像の生成



13, 000px × 13, 000pxの画像



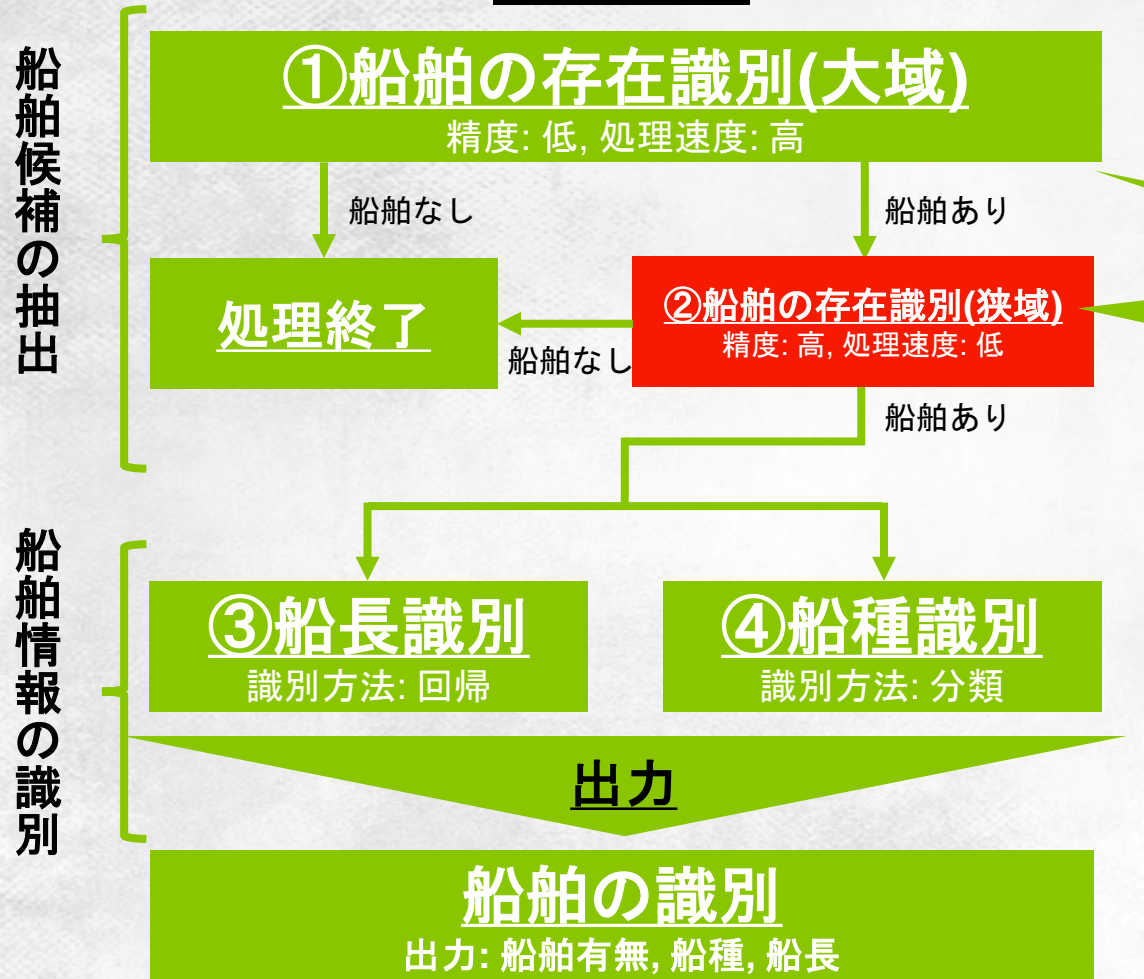
閾値によって船舶検出の感度を変更可能

Threshold = 0.7

船舶識別のアプローチ



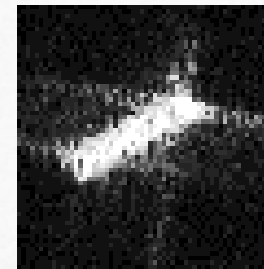
全体像



船舶の存在識別 (狭域)

300枚/s程度
精度: 100%

分けることで
120倍高速化



IF

船舶 → ③,④へ進む

波、~~ノイズ~~

処理を停止

大域で船舶を識別後、狭域で精度の高い船舶識別をかけていく

船舶識別のアプローチ



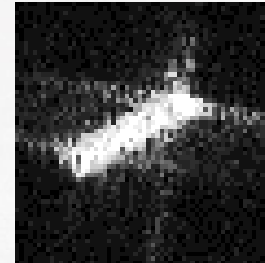
全体像

船舶候補の抽出

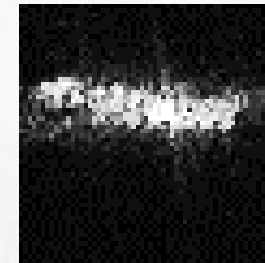
船舶情報の識別



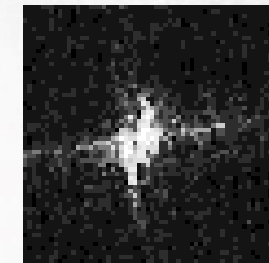
100枚1秒程度



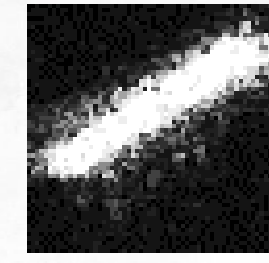
AIS: 182m
予測: 209m



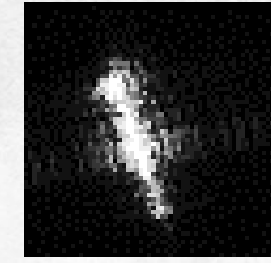
AIS: 263m
予測: 195m



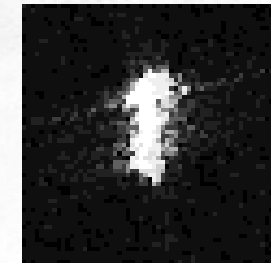
AIS: 55m
予測: 59m



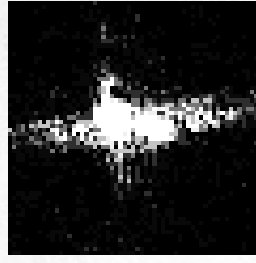
AIS: 328m
予測: 352m



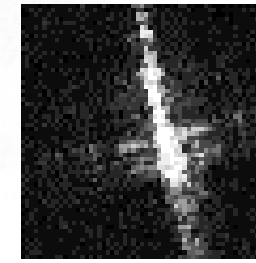
AIS: 244m
予測: 238m



AIS: 140m
予測: 103m



AIS: 96m
予測: 95m



AIS: 74m
予測: 103m

船長の識別

誤差の分散13%、中央値17%

船舶識別のアプローチ



全体像

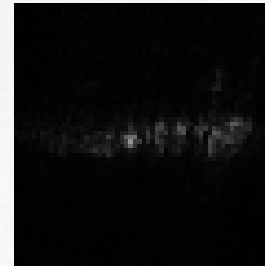
船舶候補の抽出

船舶情報の識別

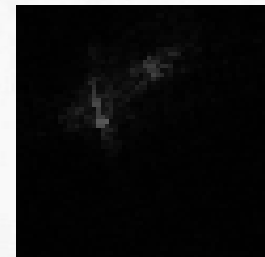


精度: 66%程度
340枚/s程度

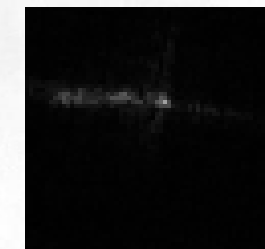
船種の識別



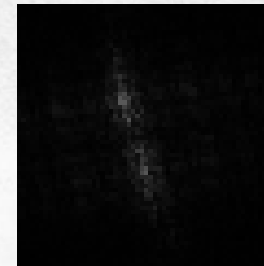
Fish. : 0%
Pass. : 0%
Cargo: 98%
Tanker: 2%



Fish. : 0%
Pass. : 0%
Cargo: 100%
Tanker: 0%



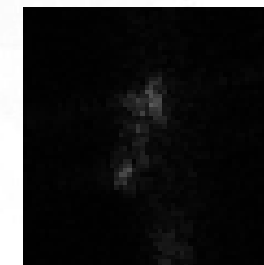
Fish. : 0%
Pass. : 0%
Cargo: 9%
Tanker: 91%



Fish. : 0%
Pass. : 33%
Cargo: 47%
Tanker: 19%



Fish. : 0%
Pass. : 0%
Cargo: 56%
Tanker: 44%



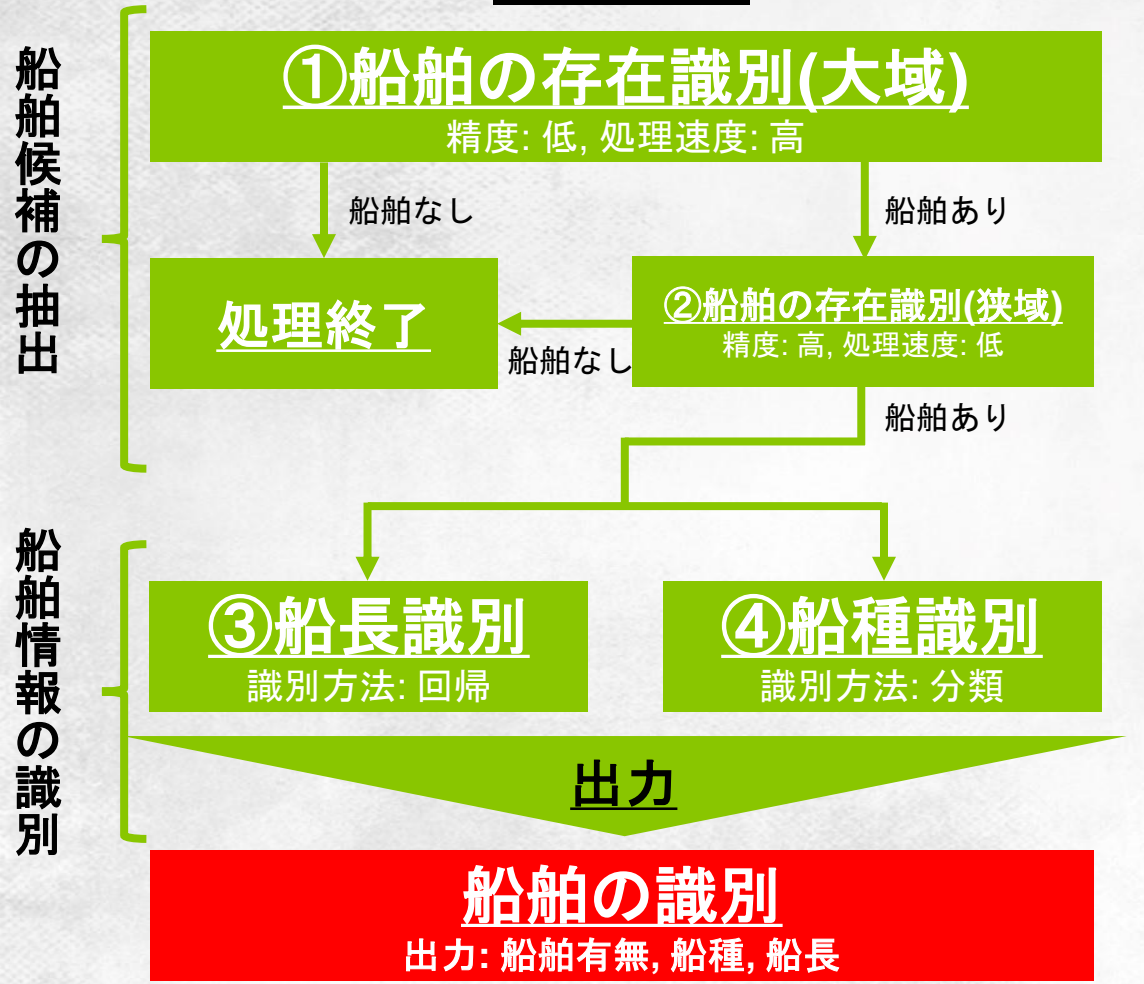
Fish. : 0%
Pass. : 0%
Cargo: 19%
Tanker: 81%

精度は高くないが、それでも人間の限界は超えている

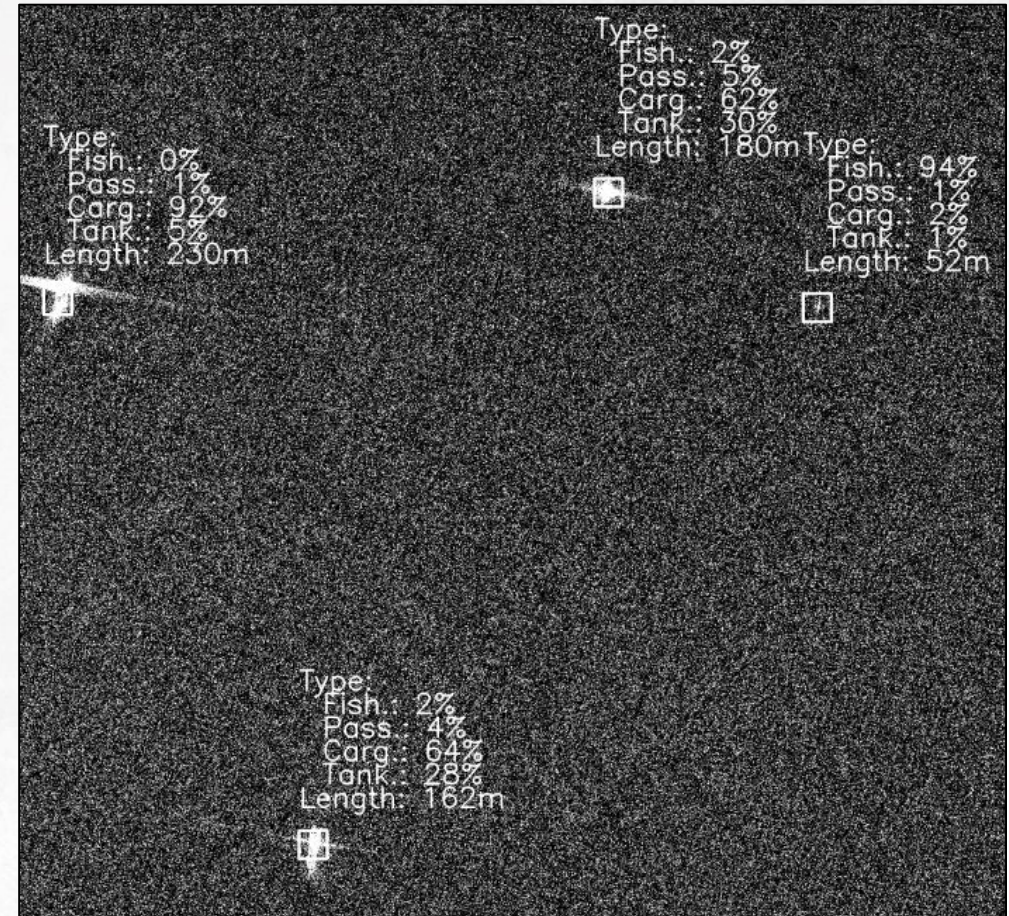
船舶識別のアプローチ



全体像

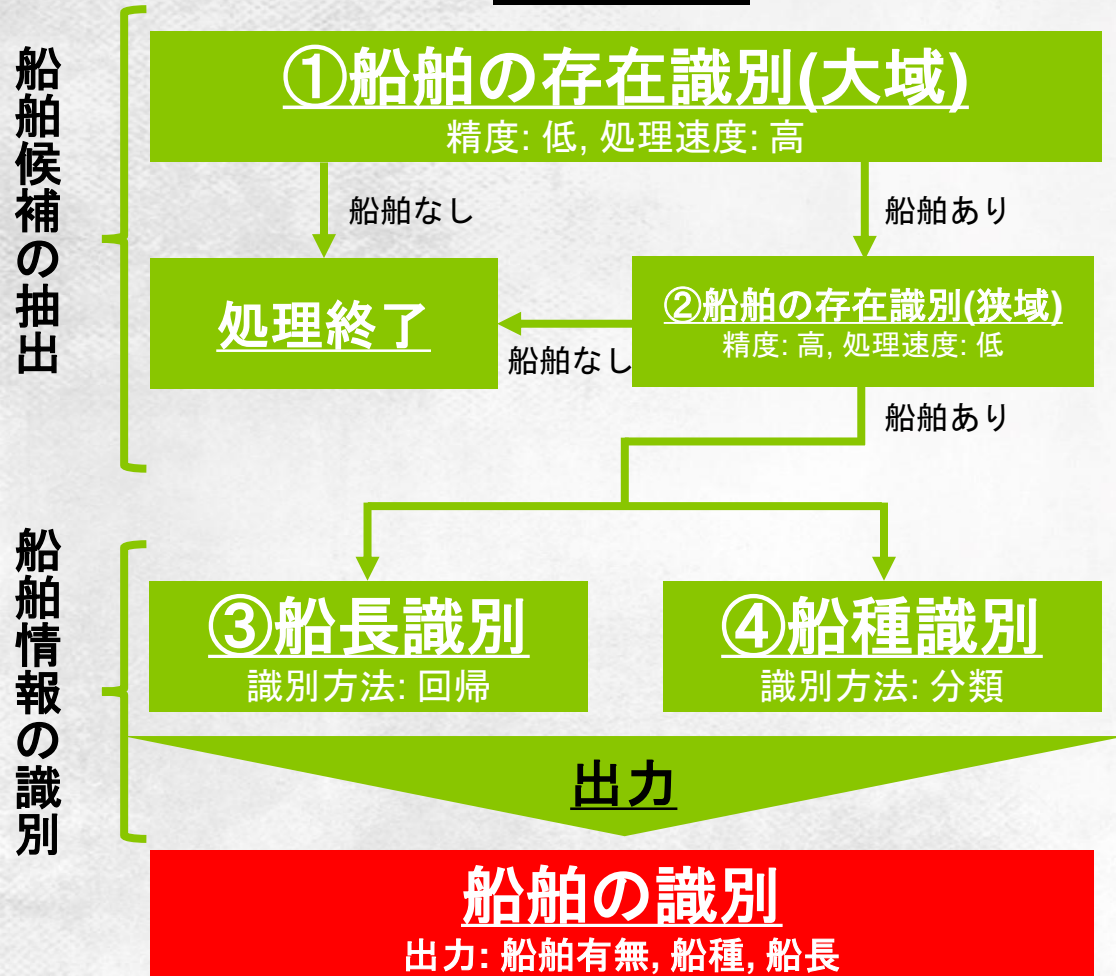


最終出力 (例)



船舶識別のアプローチ

全体像



最終出力 (例)



サイドローブの影響を除去して船舶を抽出できている。これがCNNの強み。



Ⅲ Evaluation

識別評価

船舶の識別（大域）

■ 未学習データ513隻を識別した結果

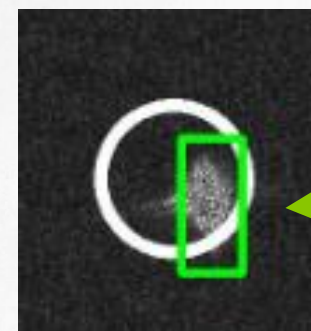
	船舶数	識別率
AISの船舶	513	99.8%
識別数	512	

Threshold = 0.7

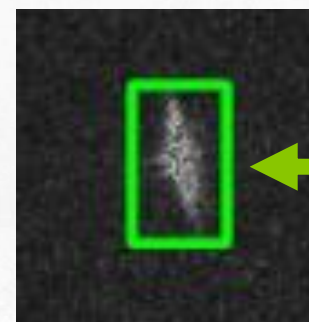
■ 船舶の検出精度の算出

- 識別率 = $\frac{\text{AISの船舶を識別した数}}{\text{AISの船舶数}}$
 - 船舶 (T) を船舶 (P) と言っている数しかわからない
 - 真値 (F) が不明のためFPは計算できない

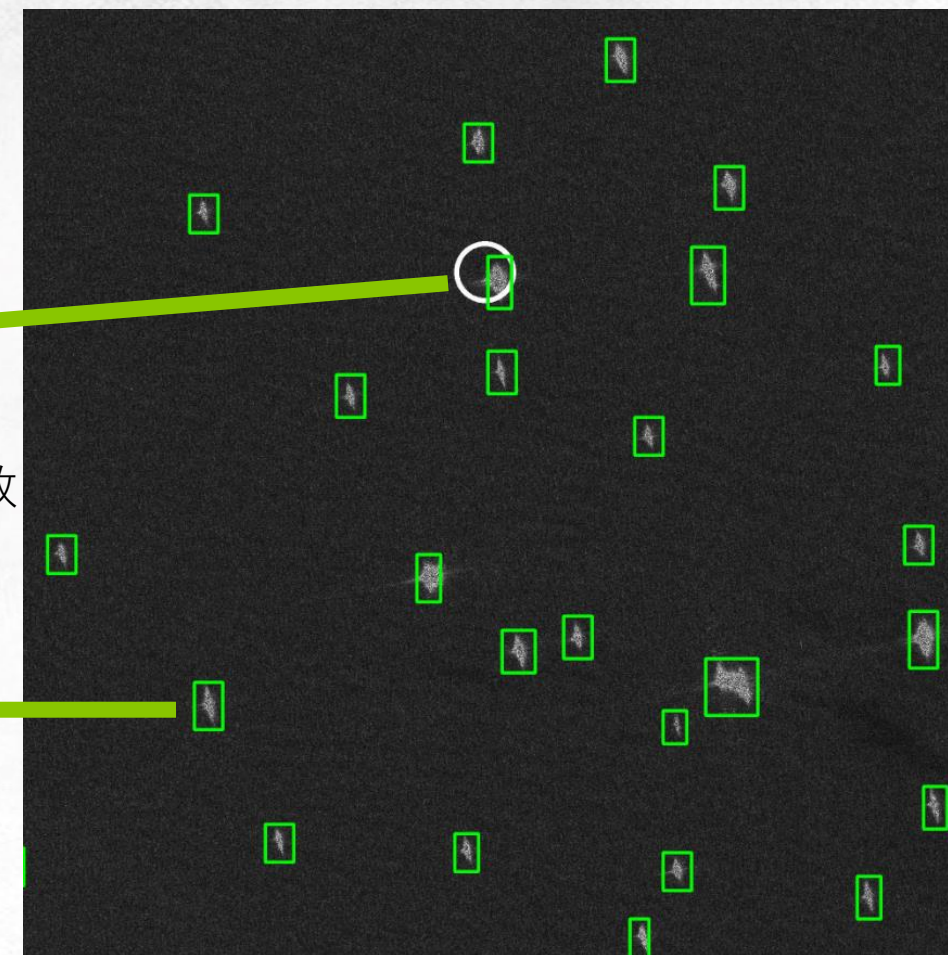
		真値	
		True (T)	False(F)
予測結果	Positive (P)	TP (True Positive)	FP (False Positive)
	Negative (N)	FN (False Negative)	TN (True Negative)



AISと識別が合致



識別のみ
(船舶か不明)



船舶識別(矩形: Deep Learning, 円形: AIS)

船舶の識別（大域）

■ 未学習データ513隻を識別した結果

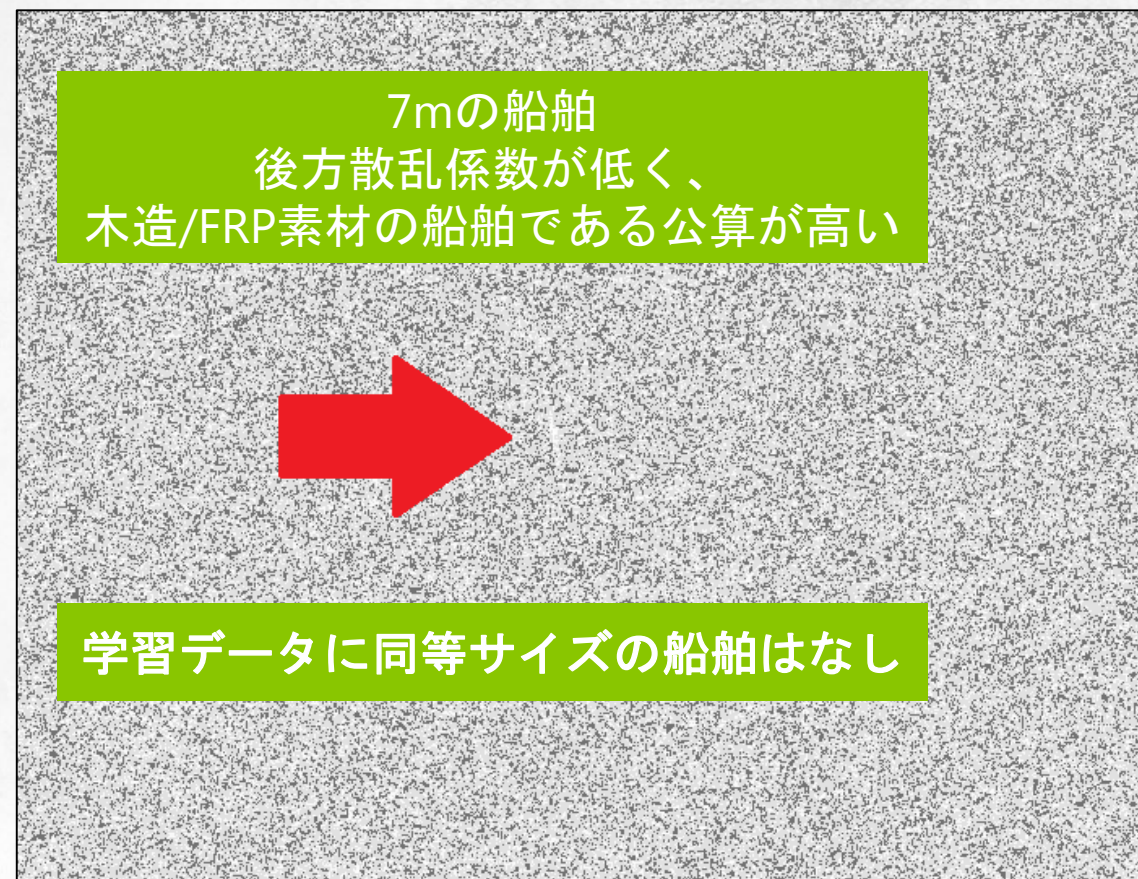
	船舶数	識別率
AISの船舶	513	99.8%
識別数	512	

Threshold = 0.7

■ 船舶の検出精度の算出

- 識別率 = $\frac{\text{AISの船舶を識別した数}}{\text{AISの船舶数}}$
 - 船舶 (T)を船舶 (P)と言っている数しかわからない
 - 真値 (F)が不明のためFPは計算できない

		真値	
		True (T)	False(F)
予測結果	Positive (P)	TP (True Positive)	FP (False Positive)
	Negative (N)	FN (False Negative)	TN (True Negative)



識別できなかった船舶

船舶の識別（狭域）



未学習データ513隻を識別した結果

		真値	
		船舶	ノイズ
識別	船舶	513	2
	ノイズ	0	511

F-measure

Overall Accuracy	99.8% (100% if only ships)
Precision	99.6%
Recall	100%
F-measure	99.8%

F-measure

		真値	
		True (T)	False(F)
予測結果	Positive (P)	TP (True Positive)	FP (False Positive)
	Negative (N)	FN (False Negative)	TN (True Negative)

$$\text{Overall accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

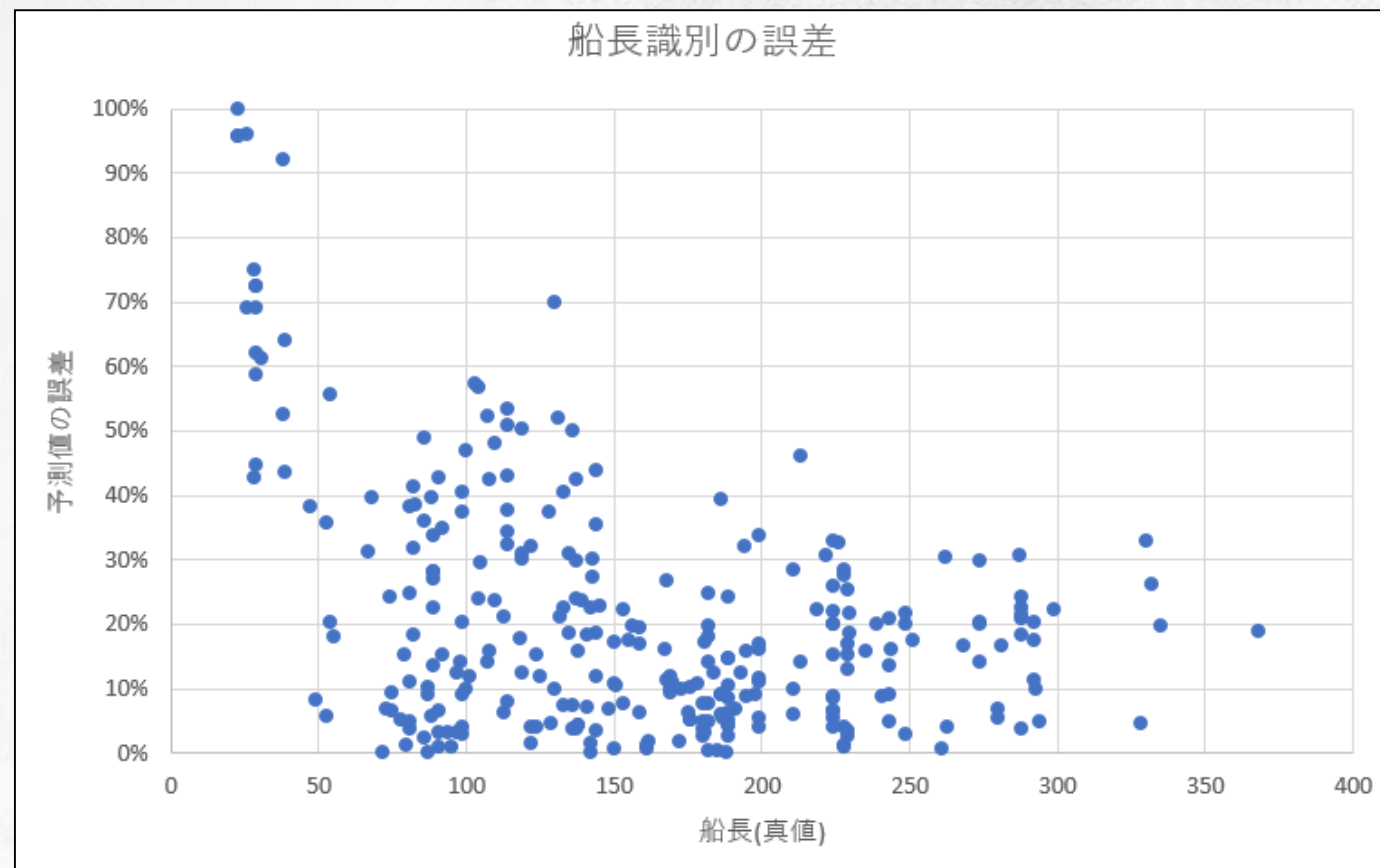
$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$F - \text{measure} = \frac{2\text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}}$$

誤差表

	誤差
分散値	13%
中央値	17%
平均値	26%



未学習データ299隻の船長を識別した際の誤差散布図

船種の識別



		クラス				
		Fishing	Passenger	Cargo	Tanker	
識別結果	Fishing	25	0	9	0	73.5%
		7.3%	0.0%	2.6%	0.0%	26.5%
	Passenger	1	1	13	1	6.3%
		0.3%	0.3%	3.8%	0.3%	93.8%
	Cargo	8	1	176	12	89.3%
		2.3%	0.3%	51.6%	3.5%	10.7%
	Tanker	4	2	66	22	23.4%
		1.2%	0.6%	19.4%	6.5%	76.6%
		65.8%	25.0%	66.7%	62.9%	65.7%
		34.2%	75.0%	33.3%	37.1%	34.3%

未学習データ341隻の船種を識別した結果



IV Onboard

FPGA化の一検討

なぜFPGAの実装が必要？



■ 背景

- ・ 衛星で観測したデータは、地上間通信の帯域的制約により地上に全てのデータをダウンリンクできていないため、衛星側で必要なデータを取捨選択する必要がある
- ・ 衛星では消費電力の制約上ミリワットオーダ級(Total Systemとしてではない)の計算機しか余剰として積めない
- ・ CNNの推論演算をCPU vs GPU vs FPGAで比較するとFPGAの電力効率がずば抜けてよい

他のプラットフォームとの比較
(VGG11をバッチサイズ1で推論)

Platform	CPU	GPU	FPGA
Device	ARM Cortex-A57	Maxwell GPU	Zynq7020
Clock Freq.	1.9 GHz	998 MHz	143.78 MHz
Memory	16 GB eMMC Flash	4 GB LPDDR4	4.9 Mb BRAM
レイテンシ [ms]	4210.0	27.23	2.37
FPS [s^{-1}]	(0.23)	(36.7)	(421.9)
Power [W]	7	17	2.3
電力効率 [FPS/W]	0.032	2.2	182.6

[10] Nakahara Hiroki, "A trend of the Deep Learning Research and Development for an FPGA", 人工知能 Vol.33 No.7

FPGAによる実装の概要

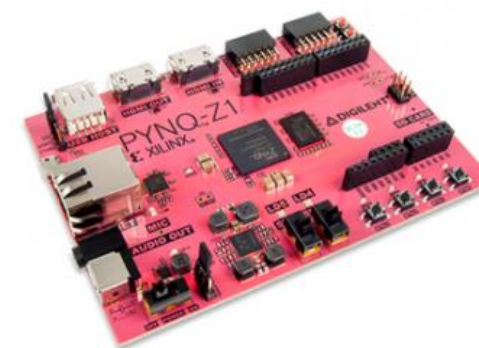


■ 概要

- ・ 船舶検出アルゴリズムの内、②船舶の存在識別(狭域)をFPGAボード上に実装
- ・ 開発効率化のため、PYNQ-Z1のボード(性能がRTG-4の3分の1程度)とライブラリを使用
 - CNNの処理が最適化されているが、制約として既にFPGAのボードに焼かれた回路を使うことになる
- ・ 学習フェーズをGPUで行いWeightを算出、そのWeightを基に推論フェーズをFPGAで動作させた
- ・ 船舶は既存のネットワークに入力サイズを合わせるために圧縮(Bilinear)を掛けている

■ 環境

- ・ ボード: PYNQ-Z1 (ZYNQ XC7Z020-1CLG400C)
- ・ モデル: CNV & Binarized Neural Network
 - 入力: 28pixels × 28pixels, 出力: 2classes
 - Conv1 (3, 3, 1, 64) -> Conv2 (3, 3, 1, 64) -> Pool3 (Max_2x2) -> Conv4 (3, 3, 1, 128) -> Conv5 (3, 3, 1, 128) -> Pool6 (Max_2x2) -> Conv7 (3, 3, 1, 256) -> Conv8 (3, 3, 1, 256) -> Fc9((7 * 7 * 256), 512) -> Fc10(512, 512) -> Fc11(512, 2)



Conv: Convolution

Pool: Pooling

Fc: Fully Connected

Conv (Kernel_X, Kernel_Y, Channel, Output)

Fc (Input, Output)

Max_2x2: MaxPooling. kernelSize(2x2), stride(2x2)

FPGAによる船舶の存在識別(狭域)の結果



■ 識別の結果

- ・オリジナルで実装したCNN (based on VGG19)よりCNV & Binarized Neural Networkの方がノイズに対して精度が高くなった

CNV & Binarized Neural Network

		真値	
		船舶	ノイズ
識別	船舶	513	0
	ノイズ	0	513

入力画像の圧縮によってノイズへの耐性が強くなったと考えられる

Original Network (CNN)

		真値	
		船舶	ノイズ
識別	船舶	513	2
	ノイズ	0	511

- ・精度が高くなった理由として、BNN化したことによるものではなく入力画像を圧縮したことによる特徴量の削減でノイズも少なくなったためだと考えられる
 - CNNのネットワークにおいても入力画像を圧縮することで同様にノイズへの耐性が強くなったことが分かった
 - ノイズに比べて船舶の特徴量はスケールが大きいので、画像を圧縮されてもノイズより最後まで残る
- ・圧縮することによって小型船舶の特徴量が消滅する可能性があるため、ノイズの除去率を減らすことよりも、Recall率(正を正という率)を高めることを優先し、オリジナルのネットワーク構成のまがよいと判断した

FPGAによる船舶の存在識別(狭域)の結果



■ 処理速度の結果

- ・ 初期化処理やメモリ転送などのオーバーヘッドを加味し、513枚の画像を一括で推論した

1枚あたりの推論時間

Hardware	CPU (i7-5930K)	GPU (Titan X)	FPGA (Zynq-7000)
processing time of one image	7, 381 [μs]	1, 165 [μs]	330 [μs]

- ・ FPGAはCPUと比較し約22倍、GPUと比較し約4倍、高速なことが分かった。

■ LUTs/メモリ使用率

- ・ LUTs/メモリに空きはあり、大きいネットワークでもBNNならFPGAに搭載できることが分かった

Site Type	使用済み/使用可能	使用率
Slice LUTs	30, 605/53, 200	57.53%
RAMB18	72/280	25.71%

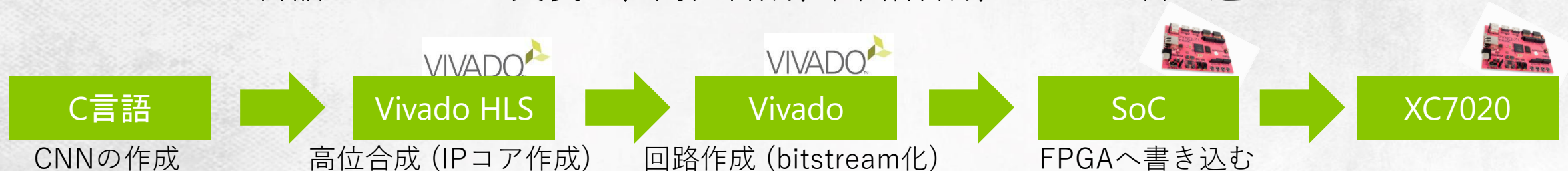
※Conv1 (3, 3, 1, 64) -> Conv2 (3, 3, 1, 64) -> Pool3 (Max_2x2) -> Conv4 (3, 3, 1, 128) -> Conv5 (3, 3, 1, 128) -> Pool6 (Max_2x2) -> Conv7 (3, 3, 1, 256) -> Conv8 (3, 3, 1, 256) -> Fc9((7 * 7 * 256), 512) -> Fc10(512, 512) -> Fc11(512, 2)

FPGAによるオリジナルCNNの実装



■ 実装概要

- PrimitiveなC言語にてCNNを実装し、高位合成、回路作成、FPGAへ書き込む



■ 処理速度

- 入力画像28pixels × 28pixels(Dataset: Mnist)を推論するのに **1枚あたり45,000[μ s]**かかった
 - Conv1 (3, 3, 1, 8) -> Pool2 (Max_2x2) -> Conv3 (3, 3, 8, 16) -> Pool4 (Max_2x2) -> Fc5((7 * 7 * 16), 32) -> Fc6(32, 10)
- C言語から高位合成を行うと1ステップ(1 cycle)がFPGAの固定クロックで動作するため非常に遅い
 - 複雑な条件分岐や処理を取り除き、かつ並列化する必要がある
- ハードウェア記述言語(HDL)で開発しなければ何れ限界を迎える

FPGAによるオリジナルCNNの実装



■ メモリ使用率について

- ・ 精度は浮動小数点float (32bit)を使用
- ・ メモリ使用率は46%となった

Site Type	使用済み/使用可能	使用率
Slice LUTs	17, 001/53, 200	31%
RAMB18	130/280	46%

- ・ 入力層(1, 2)と結合層(7)で80%ものメモリを占めている。
- ・ 6層あたりからは層をいくら深くしてもメモリには収まりそう
- ・ 結合層では画像サイズ×フィルタ×ユニット数の量に応じて配列が必要になるため、大きくなる。このため画像サイズを小さくする必要がある。
- ・ 入力層と結合層を工夫すれば、かなり大きめの層もバイナライズ化せずともFPGAに乗りそう

→しかし、XOR回路で単純計算できないため処理速度が遅くなる

各層	層の名前	要素数	割合	役割
	1 Conv	784	2%	input
		72	0%	weight
		6,272	13%	output
	2 Relu	6,272	13%	output
	3 Pool	1,568	3%	output
	4 Conv	1,152	2%	weight
		3,136	6%	output
	5 Relu	3,136	6%	output
	6 Pool	784	2%	output
	7 Fc	25,088	52%	weight
		32	0%	output
	8 Relu	32	0%	relu
	9 Fc	320	1%	weight
		10	0%	output
	10 Softmax	10	0%	output
		48,668		33,688

プログラム内で保持される主な変数

FPGAの実装に関するまとめ



■ FPGAは有用か？

- ・ 推論に用いる場合においては、CPU及びGPUに対して圧倒的な性能さを確認できた
- ・ BNN化した場合においても十分な精度を確認できた
- ・ FPGAフレンドリな方法で実装をしなければ、全く性能がでないことを確認できた

■ FPGAのメモリ問題

- ・ BNN化することによって様々なサイズのモデルに対応することができる
- ・ BNN化せずとも入力層及び結合層を工夫することで大きなサイズのモデルにも対応できそうである
- ・ BNN化する/しないに関わらず大きな入力画像を扱うことが困難であることが確認できた
 - 入力画像に関してはバイナライズ化しないため、そのままの大きさをFPGAに乗ることになる
 - GPUのように1000 pixels × 1000 pixelsをそのまま入力画像として扱うようなことはできない



Summary

まとめ、展望

- SAR画像から船舶位置(存在)、船長、船種を識別した
- 船舶識別では既存の手法(CFARやHaar-Cascadesなど)と比較し、CNNが強力であることを確認した
- 船舶の位置(存在)識別精度は99.8%であった
- 船長の識別精度は分散13%(誤差)であった
- 船種の識別精度は65.7%であった
- FPGAで船舶識別を実装できそうであるが、処理速度及び精度を出すためにはFPGA特有の工夫が必要