KYOTO 2022
IEEE/RSJ International Conference on Intelligent Robots and Systems

CA-SpaceNet: Counterfactual Analysis for 6D Pose Estimation in Space

Shunli Wang[†], Shuaibing Wang[†], Bo Jiao, Dingkang Yang, Liuzhen Su, Peng Zhai, Chixiao Chen[*], Lihua Zhang[*]

Academy for Engineering & Technology, Fudan University

Code: https://github.com/Shunli-Wang/CA-SpaceNet
Video: https://www.youtube.com/watch?v=h-vzCdersVQ

[†] Equal contribution. [*] Corresponding authors.

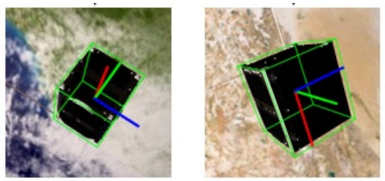
Outline

- 1. Background & Motivations
- 2. Proposed CA-SpaceNet
- 3. Experimental Results
- 4. Conclusion

Outline

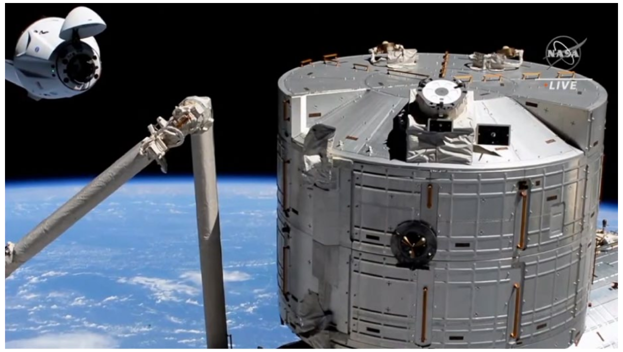
- **1. Background & Motivations**
- 2. Proposed CA-SpaceNet
- 3. Experimental Results
- 4. Conclusion

1. Background & Motivations



Reliable and stable 6D pose estimation algorithms

Basic technology



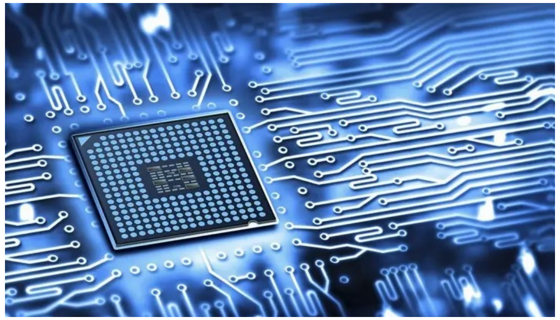
SpaceX Crew-2 Docking Mission

Preview of the ClearSpace-1

Challenges in the space scene:

Harsh imaging conditions

Background interference

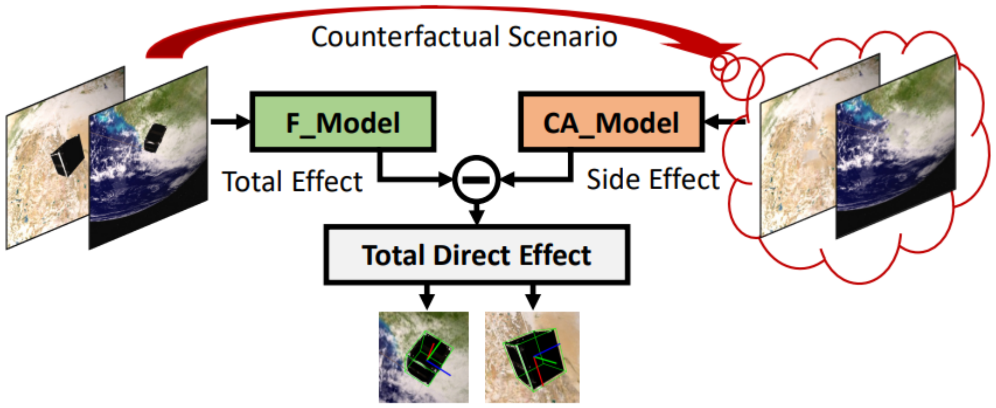


Limited power consumption

1. Background & Motivations

Complex Background Interference

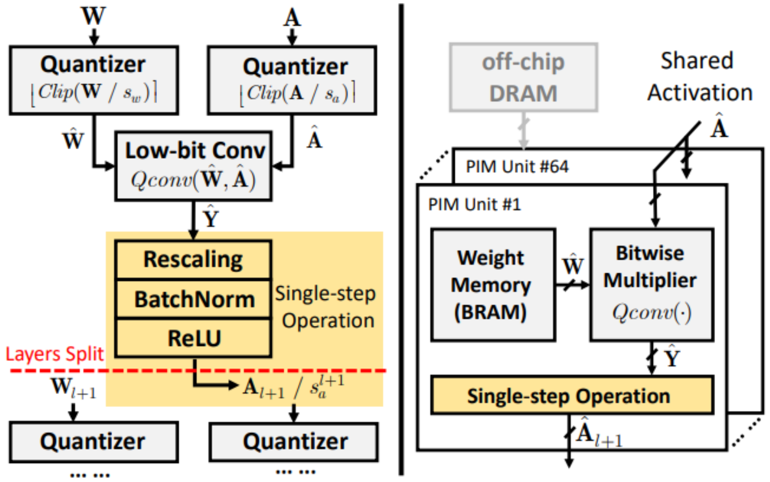
How can we remove harmful features from the composite features?



Main idea of the proposed CA-SpaceNet.

Limited Computing Resources & Power Consumption

How can we achieve efficient deployment of network models in low-power scenarios?

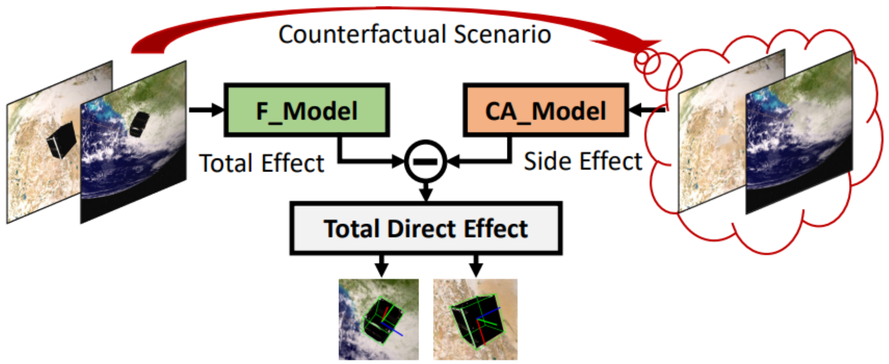


Quantization inference and the PIM accelerator architecture.

Outline

- 1. Background & Motivations
- **2. Proposed CA-SpaceNet**
- 3. Experimental Results
- 4. Conclusion

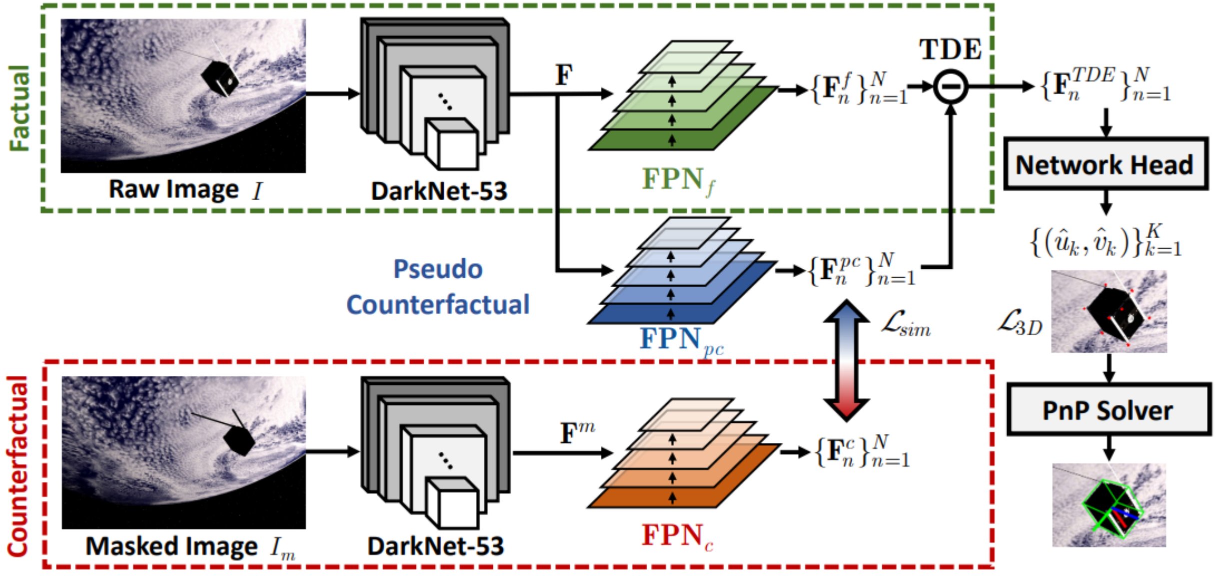
2. Proposed CA-SpaceNet



Counterfactual thinking

Feature Extraction → Feature Aggregation → Network Head → PnP Solver

Basic components of a two-stage 6D pose estimator.



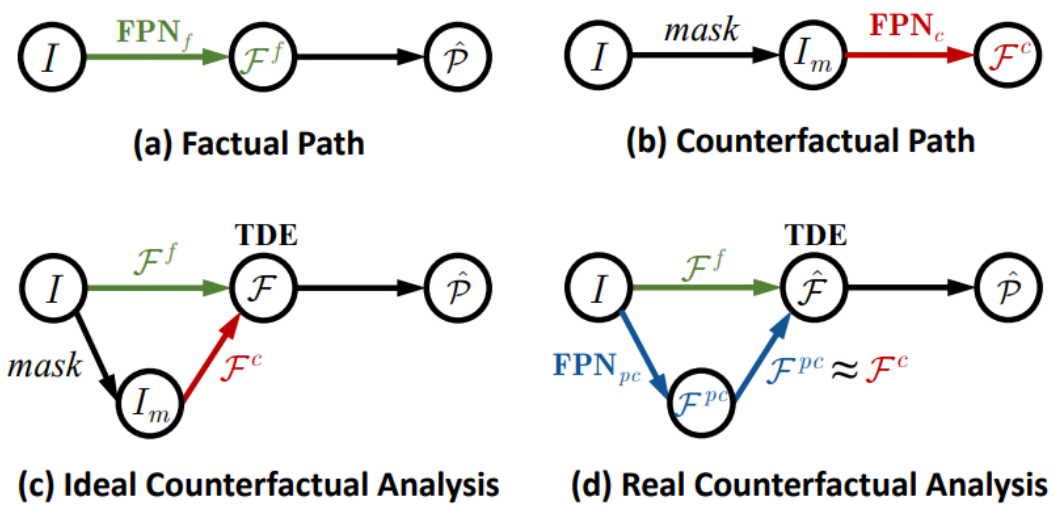
Actual network structure (CA-SpaceNet)

Factual path: Extract the feature of the whole image.

Counterfactual Path: Extract the feature of the background information.

Pseudo Counterfactual Path: Mimic the output feature of the counterfactual path.

2. Proposed CA-SpaceNet



Simplified causal graphs of the CA-SpaceNet in four situations.

Feature Extraction → Feature Aggregation → TDE Calculation → Network Head → PnP Solver

Five stages of the CA-SpaceNet.

Factual path: $\mathcal{F}^f = \mathbf{FPN}_f(\mathbf{F})$

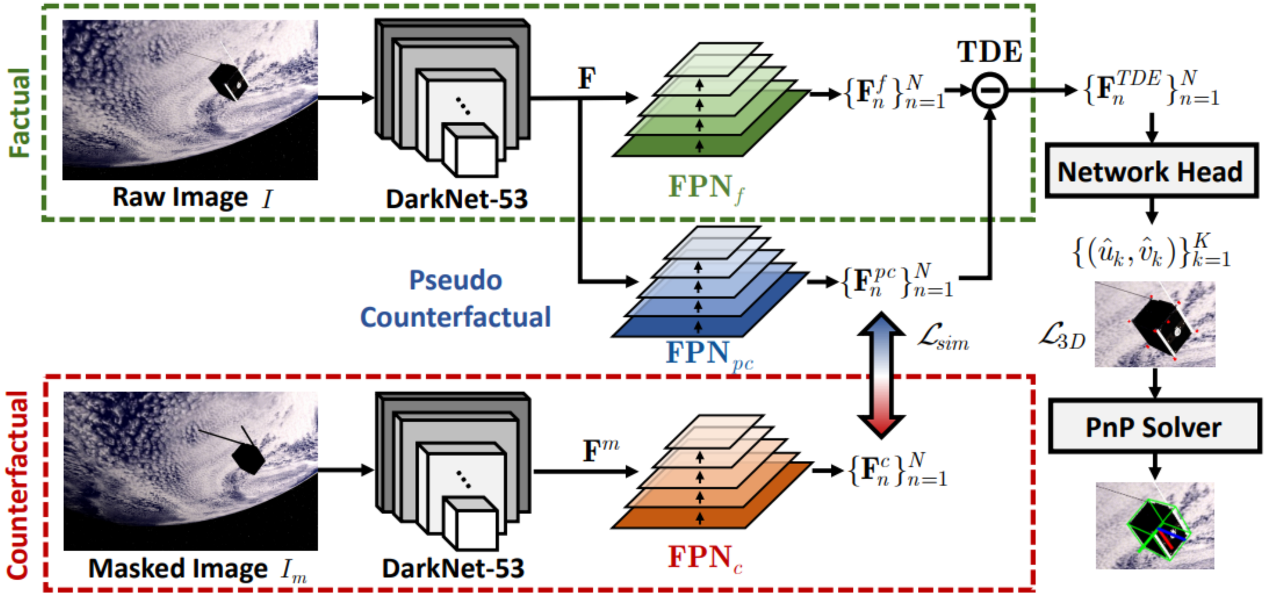
Counterfactual Path: $\mathcal{F}^c = \mathbf{FPN}_c(\mathbf{F}^m)$

Pseudo Counterfactual Path: $\mathcal{F}^{pc} = \mathbf{FPN}_{pc}(\mathbf{F})$

$\mathcal{L}_{sim}$

Total Direct Effect (TDE): $\begin{cases} \mathcal{F} = \mathcal{F}^f - \mathcal{F}^c \\ \hat{\mathcal{F}} = \mathcal{F}^f - \mathcal{F}^{pc} \end{cases}$

2. Proposed CA-SpaceNet



Overview of the proposed CA-SpaceNet.

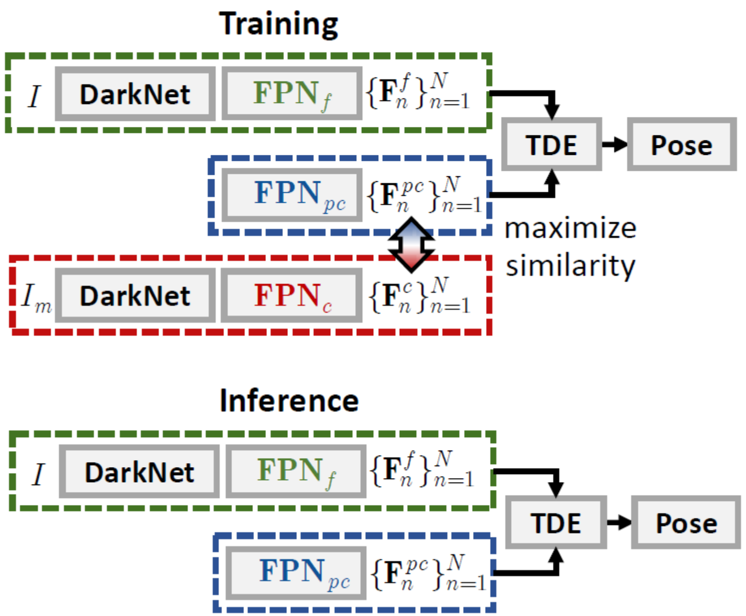


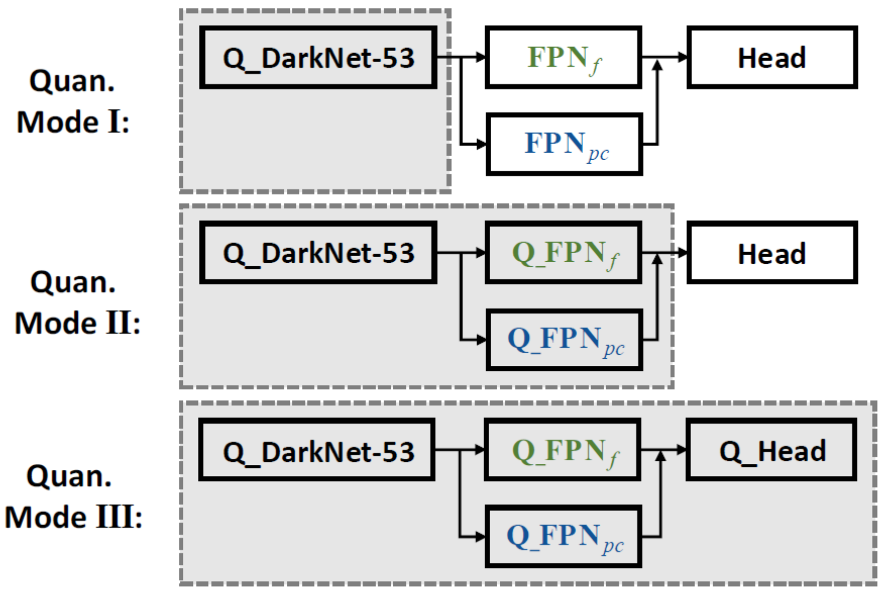
Diagram of training and inference phases.

- **Training**: $\mathbf{FPN}_{pc}$ will imitate $\mathbf{FPN}_c$ by maximizing the similarity between $\{\mathbf{F}_n^{pc}\}_{n=1}^N$ and $\{\mathbf{F}_n^c\}_{n=1}^N$.

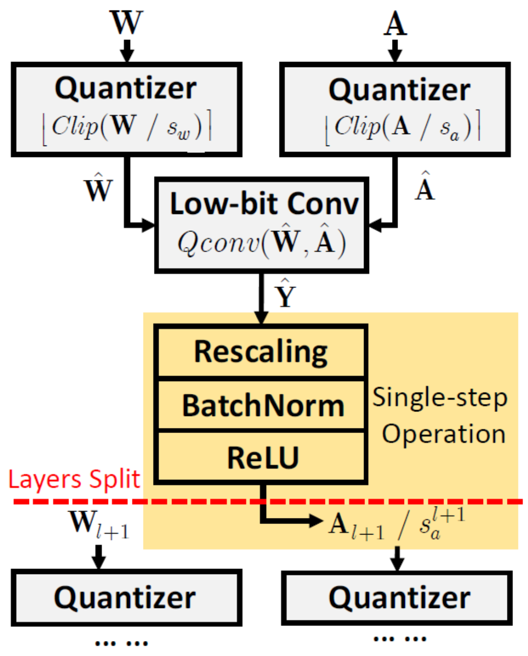
  Total Loss: $\mathcal{L} = \lambda_{3D}\mathcal{L}_{3D} + \lambda_c\mathcal{L}_{cls} + \lambda_s\mathcal{L}_{sim}$

- **Testing**: the whole counterfactual path will be removed during inference.

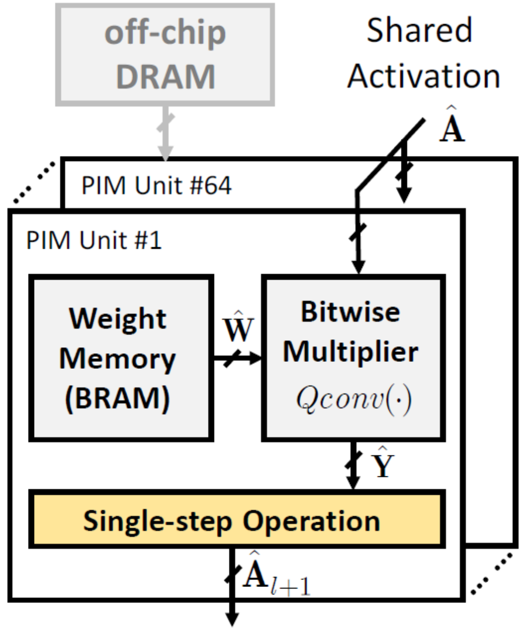
2. Proposed CA-SpaceNet



Three different quantization modes.



Quantization inference.



The PIM accelerator architecture.

Quantized weights: $\hat{\mathbf{W}} = \lfloor clip(\mathbf{W}/s_w) \rceil$

Quantized activations: $\hat{\mathbf{A}} = \lfloor clip(\mathbf{A}/s_a) \rceil$

Low-bit-width convolution: $\hat{\mathbf{Y}} = Qconv(\hat{\mathbf{W}}, \hat{\mathbf{A}})$

Dequantization: $\mathbf{Y} = \hat{\mathbf{Y}} * s_w * s_a$

Merge BN layer:

$$\begin{aligned} \mathbf{Y}^{bn}_{(i,:,:,:)} &= \frac{\mathbf{Y}_{(i,:,:,:)} - \boldsymbol{\mu}_i}{\boldsymbol{\sigma}_i} \gamma_i + \boldsymbol{\beta}_i \\ &= \frac{\gamma_i}{\boldsymbol{\sigma}_i} * s_w * s_a * \hat{\mathbf{Y}}_{(i,:,:,:)} - \frac{\boldsymbol{\mu}_i \gamma_i}{\boldsymbol{\sigma}_i} + \boldsymbol{\beta}_i \end{aligned}$$

Outline

- 1. Background & Motivations
- 2. Proposed CA-SpaceNet
- **3. Experimental Results**
- 4. Conclusion

3. Experimental Results

TABLE I

COMPARISON WITH STATE-OF-THE-ARTS ON SWISSCUBE.

Method	Near ↑	Medium ↑	Far ↑	All ↑
SegDriven [39]	41.1	22.9	7.1	21.8
SegDriven-Z [39]	52.6	45.4	29.4	43.2
DLR [5]	63.8	47.8	28.9	46.8
WDR [4]	65.2	48.7	31.9	47.9
WDR* [4]	**92.37**	84.16	61.27	78.78
CA-SpaceNet	91.01	**86.32**	**61.72**	**79.39**

TABLE II

COMPARISONS OF THE RE-TRAINING WDR* MODEL AND THE CA-SPACENET.

Method	Near ↑	Medium ↑	Far ↑	All ↑
WDR* [4]	**92.37**	84.16	61.27	78.78
WDR* [4] *w.* 30-Ep.	89.93 (-2.44)	82.09 (-2.07)	56.50 (-4.77)	75.76 (-3.02)
CA-SpaceNet	91.01 (-1.36)	**86.32** (+2.16)	**61.72** (+0.45)	**79.39** (+0.61)

TABLE III

COMPARISON WITH STATE-OF-THE-ARTS ON SPEED

Method	$\mathbf{e}_q + \mathbf{e}_t \downarrow$
SLAB Baseline [3]	0.0626
pedro-fairspace [42]	0.0571
WDR [4]	**0.0180**
WDR* [4]	0.0400
CA-SpaceNet	0.0385

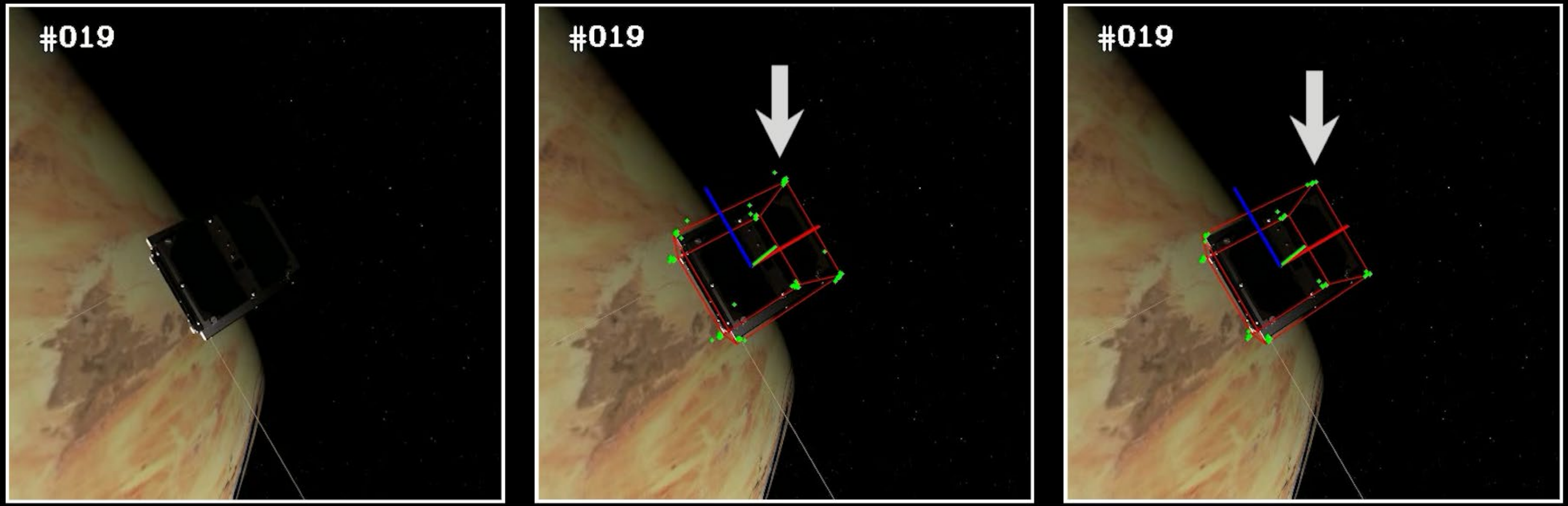
- The CA-SpaceNet can eliminate the interference of background through counterfactual analysis under the *Medium* and *Far* setting.
- The performance improvement is brought by the counter-factual analysis strategy rather than the additional 30 training epochs.

WDR*: note that the results of WDR on SPEED is obtained by the original paper without source code. We reproduced this model and reported our results as WDR*.

3. Experimental Results

testing_seq_000471

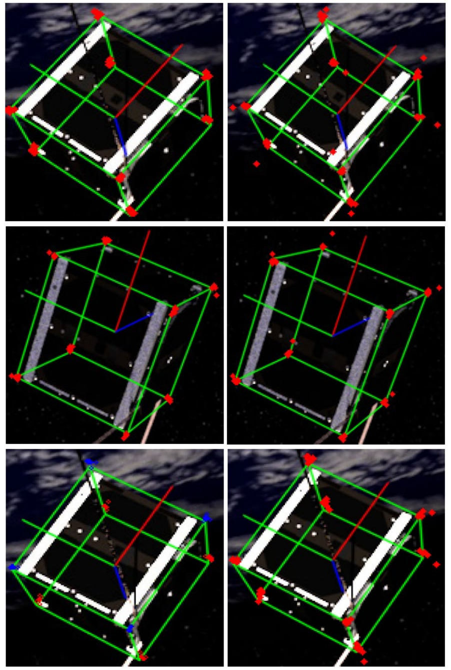
(Red boxes denote ground-truth pose. Green points denote prediction corners.)



Raw Video | WDR* | CA-SpaceNet

- In WDR*, the background interference makes the prediction points largely offset from GT corners.
- However, with the help of the causal inference strategy, the CA-SpaceNet successfully handles these complex situations.

Results of CA-SpaceNet | Results of WDR*



The CA strategy is able to weaken the adverse impact of background interference to the final results.

3. Experimental Results

TABLE IV

RESULTS ON THREE DIFFERENT QUANTIZATION MODES OF 8-BIT AND 3-BIT CA-SPACENET ON SWISSCUBE

#Bits	Quan. Mode	ADI-0.1d ↑	OPs & FLOPs	Perc.(%)
8	I	76.21	36.91 GOPs + 33.79 GFLOPs	52.21
	II	75.04	44.51 GOPs + 26.19 GFLOPs	62.96
	III	74.65	70.47 GOPs + 0.23 GFLOPs	99.67
3	I	75.10	36.91 GOPs + 33.79 GFLOPs	52.21
	II	74.47	44.51 GOPs + 26.19 GFLOPs	62.96
	III	68.68	70.47 GOPs + 0.23 GFLOPs	99.67

TABLE V

SUMMARY OF PARAMETER STORAGE SIZE

Format	#Para.	Model Size	Stor. Saving (%) ↑
FP32	51.29 M	205.17 MB	0.00
8-bit	51.29 M	51.29 MB	75.00
3-bit	51.29 M	19.23 MB	90.63

TABLE VI

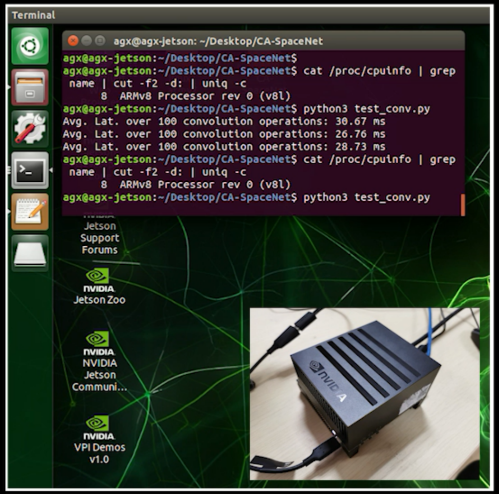
MEASURED LATENCY ON DIFFERENT HARDWARE

Device	Latency (ms) ↓
ARM v8.2 64-bit CPU (Nvidia Xavier)	26.16
Intel Core i7-8700K CPU	10.25
PIM Arch. on Ultra96v2 FPGA	**5.99**

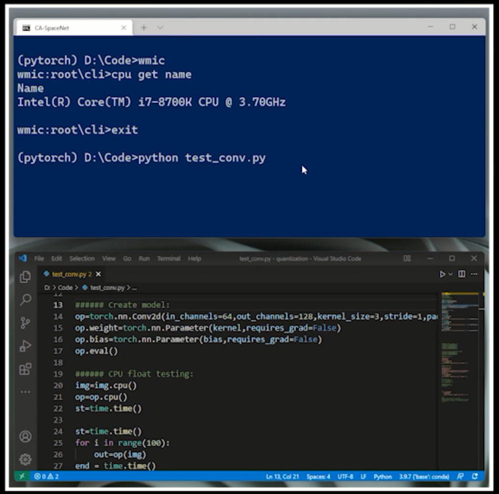
- The quantization can save a large mount of GFLOPs without significantly reducing the performance.
- Quantization greatly reduce the size of the network which makes the model easier to be deployed on devices.
- Latency test results of the PIM chip shows the high efficiency of the quantization and our actual deployment.

3. Experimental Results

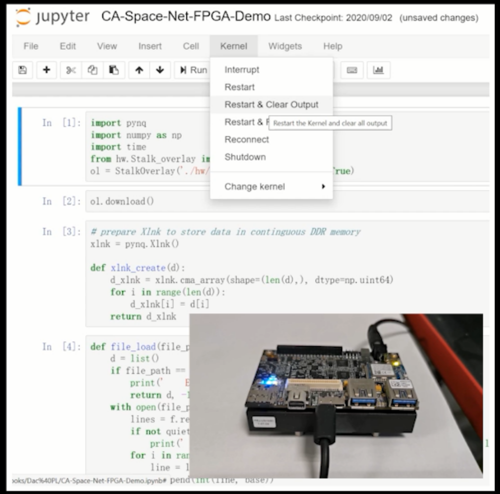
Latency Comparison



ARM v8.2 64-bit CPU
(Nvidia Xavier)



Intel Core i7-8700K CPU



Processing-In-Memory Arch.
(3-bit PIM Accelerator on Ultra96v2 FPGA)

	ARM v8.2 64-bit CPU (Nvidia Xavier)	Intel Core i7-8700K CPU	Processing-In-Memory Arch. (3-bit PIM Accelerator on Ultra96v2 FPGA)
Latency ↓	26.16 ms	10.25 ms	5.99 ms

Our deployment achieves 4.4x speedup compared with ARM v8.2 CPU and 1.7x speedup compared with Intel Core i7-8700K CPU.

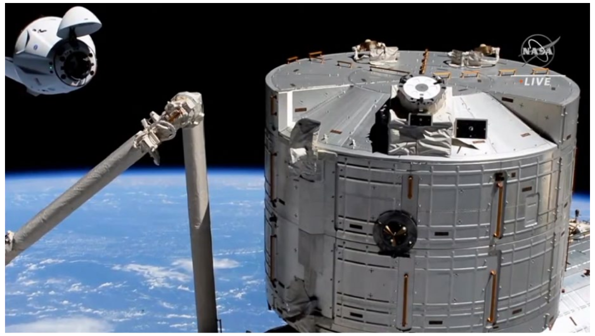
Our real deployment of the **PIM architecture** on the Ultra96v2 FPGA achieves:

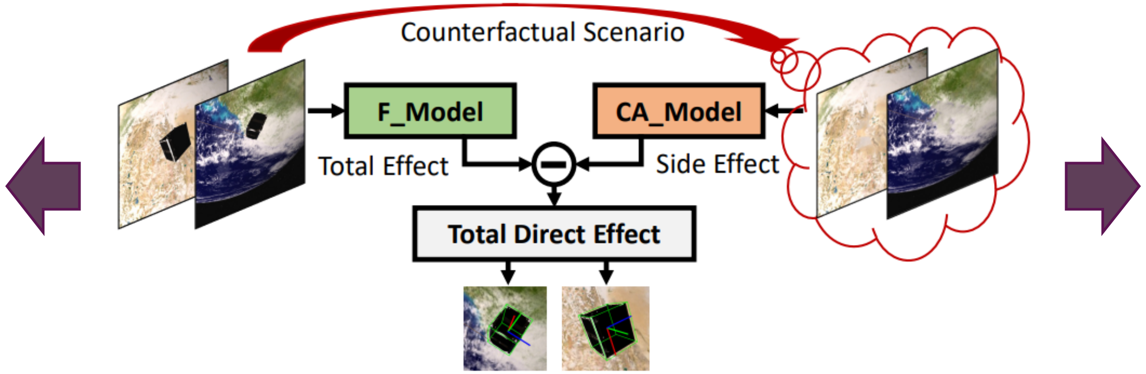
4.4x speedup than ARM v8.2 CPU,
&
1.7x speedup than Intel Core i7-8700K CPU.

Outline

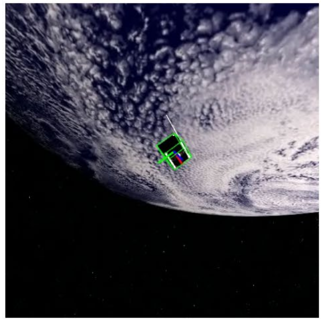
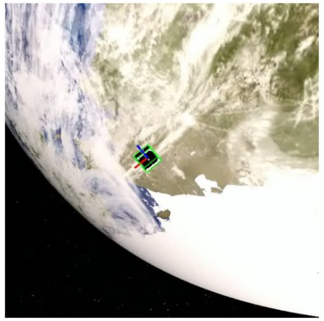
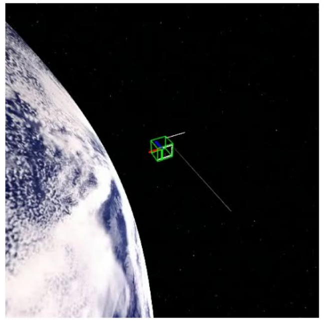
- 1. Background & Motivations
- 2. Proposed CA-SpaceNet
- 3. Experimental Results
- **4. Conclusion**

4. Conclusion

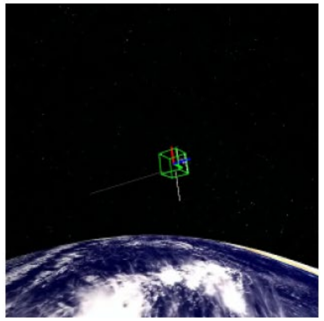
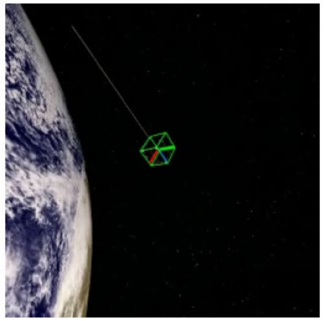
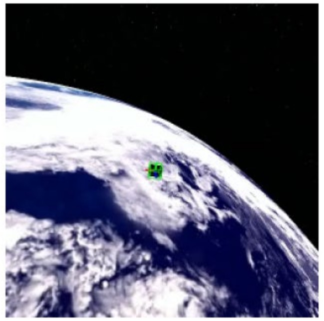




- We propose a framework named CA-SpaceNet, which is robust to the interference of complicated background information.
- Our approach outperforms state-of-the-arts on the challenging SwissCube dataset and achieves competitive results on the SPEED dataset.
- We quantize the CA-SpaceNet into a low-bit-width model and deploy a part of the quantized network into a Processing-In-Memory (PIM) chip on FPGA. Low latency proves the feasibility of our method.

Q&A

CA-SpaceNet: Counterfactual Analysis for 6D Pose Estimation in Space

Code: https://github.com/Shunli-Wang/CA-SpaceNet

Video: https://www.youtube.com/watch?v=h-vzCdersVQ