1: (weak accept)

This paper discusses the application of split computing in the domain of object detection on 3D point clouds captured by LIDAR sensors. The split computing strategy consists in performing part of the inference of a deep neural network model directly on the edge device, and the rest of the inference on an edge server. The experimental results focus on an open-source 3D object detection toolset and highlight an improvement in terms of inference time when leveraging split computing, together with an inherent gain in data security.

Overall, the paper is well-written and easy to follow, and explores a relevant domain of research related to edge computing leveraging machine learning pipelines.

The main weakness of the paper is that the novelty of the proposed approach feels limited. Specifically, split computing is a distributed computing technique which is quite general and can be applied to a wide variety of machine learning inference pipelines. Therefore, applying this method to the specific domain of 3D object detection does not constitute a strong original contribution by itself. Consider that Sec. III - Approach is very brief and for the most part simply reports how split computing works, as already adequately described in the introductory and background sections (Sections I and II) and without adding significant technical details or improvements. As a result, the main contribution would be in the evaluation of the split computing approach for the OpenPCDet 3D object detection library, i.e., the experimental results.

The experiments show promising results and demonstrate the feasibility of the approach, but only focusing on a single instance of a very specific use case.

Minor comments:

- The following paper from the previous edition of RAGE seems to be closely related to the present paper, as it deals with split computing for CNNs: "Sanchez Leal, I., Saqib, E., Shallari, I., Jantsch, A., Krug, S., & O'Nils, M. (2023). Waist Tightening of CNNs: A Case study on Tiny YOLOv3 for Distributed IoT Implementations. In Proceedings of Cyber-Physical Systems and Internet of Things Week 2023 (pp. 241-246)."

- When reading the experimental results, it seems strange at first that the inference times in Figures 6 and 7 with the split after conv2 become larger than the inference times without splitting. This is implicitly clarified later when considering the large communication times in Figure 9. I suggest explaining earlier that the large inference time is due to this extra latency.

- The experiments should report how many evaluations of the inference and transfer times were performed.

- In Page 4, clarify what "transfer data" means. It seems to refer to the data that needs to be transfered from the edge device to the edge server.

- Page 4: the text states that "if splitting after the fourth convolutional layer, the outputs of the second, third, and fourth convolutional layers need to be forwarded from edge devices to edge servers. For this reason, the performance of splitting after the third and fourth convolutional layers is clearly inferior to that of splitting after the second convolutional layer." This reasoning is not clear to me; e.g., why would the inference time be clearly inferior in that case.

- There is no need to use title case in figure captions, just follow the capitalization conventions for normal text.

My recommendation for the paper is to accept the paper, provided that the contributions are deemed sufficient by the other reviewers.

1: (弱受)

本稿では、LIDARセンサーによって取得された3D点群上の物体検出の領域における分割コンピューティングの適用について議論する。スプリットコンピューティング戦略は、ディープニューラルネットワークモデルの推論の一部をエッジデバイス上で直接実行し、残りの推論をエッジサーバー上で実行する。実験結果は、オープンソースの3Dオブジェクト検出ツールセットに焦点を当て、スプリットコンピューティングを活用した場合の推論時間の改善と、データセキュリティの本質的な向上を強調している。

全体的に、この論文はよく書かれており、分かりやすく、機械学習パイプラインを活用したエッジコンピューティングに関連する研究領域を探求している。

この論文の主な弱点は、提案されたアプローチの新規性が限定的に感じられることである。具体的には、分割コンピューティングは分散コンピューティング技術であり、非常に一般的で、様々な機械学習推論パイプラインに適用できる。従って、この手法を3次元物体検出という特殊な領域に適用しても、それだけでは強い独創的な貢献とはならない。第III章「アプローチ」は非常に簡潔であり、その大部分は、すでに入門と背景の章（第I章と第II章）で十分に説明されているように、分割コンピューティングがどのように機能するかを単に報告しているだけであり、重要な技術的詳細や改良点を追加していないことを考慮されたい。その結果、主な貢献は、OpenPCDet 3Dオブジェクト検出ライブラリに対する分割コンピューティングアプローチの評価、すなわち実験結果であろう。

実験結果は、有望な結果を示し、アプローチの実現可能性を示していますが、非常に特殊なユースケースの1つのインスタンスに焦点を当てています。

細かいコメント

- RAGEの前版に掲載された以下の論文は、CNNの分割計算を扱っているため、本論文と密接に関連していると思われる： 「Sanchez Leal, I., Saqib, E., Shallari, I., Jantsch, A., Krug, S., & O'Nils, M. (2023). CNNのウエスト引き締め： 分散IoT実装のための小さなYOLOv3のケーススタディ。In Proceedings of Cyber-Physical Systems and Internet of Things Week 2023 (pp. 241-246)".

- 実験結果を読むと、conv2の後に分割した図6と図7の推論時間が、分割していない推論時間よりも大きくなっていることが、最初は不思議に思える。これは、後で図9の通信時間が大きいことを考えると、暗黙のうちに明らかになる。推論時間が大きいのは、この余分な待ち時間のためであることを先に説明することを提案する。

- 実験では、推論時間と転送時間の評価を何回行ったかを報告すべきである。

- 4ページ目の「転送データ」の意味を明確にしてください。エッジ・デバイスからエッジ・サーバーに転送される必要のあるデータを指しているようである。

- 4ページ：本文では、「第4畳み込み層の後に分割する場合、第2、第3、第4畳み込み層の出力をエッジデバイスからエッジサーバーに転送する必要がある。このため、3層目と4層目の畳み込み層の後に分割した場合の性能は、2層目の畳み込み層の後に分割した場合よりも明らかに劣る。" この推論は私にはよくわからない。例えば、なぜその場合、推論時間が明らかに劣るのか。

- 図のキャプションに大文字と小文字を使う必要はなく、通常のテキストの大文字と小文字の規則に従ってください。

この論文に対する私の推奨は、他の査読者が十分な貢献であると判断した場合に限り、受理することである。

2: (accept)

Summary:

This paper proposes a split computing (SC) approach for 3D point cloud detection on edge devices found in applications such as autonomous driving. The proposed SC strategy aims to (1) reduce processing time, (2) reduce the computational load on edge devices, and (3) preserve the privacy of the 3D point cloud data. To achieve this, the author’s SC strategy splits an existing model (Voxel R-CNN) into a head and tail model which will be deployed and run inference on the edge device and edge server, respectively. The effects of splitting the model at different points is discussed and explored during their experimental evaluation.

Strengths:

This paper is very well written and easy to follow. Moreover, the proposed SC approach is clearly defined and well situated within the context of their target application and other existing SC work. Considering the three goals the authors establish for their system (stated in my summary and in Section I of the paper), the presented experimental evaluation soundly explores the necessary metrics (i.e., inference time, data size, and data transfer time) for determining the “success” of their approach compared to other model configurations. Given the limited space for this submission, the authors have done a good job at explaining and evaluating their approach.

Weaknesses:

The paper focuses heavily on autonomous driving technology but it does not provide any of the timing constraints for 3D point cloud detection on an edge device in such a scenario. Including some values related to these timing constraints would help contextualize the results presented in the evaluation (e.g., do all of the model configurations meet the inference speed requirements of an autonomous driving system).

The evaluation is well done but there were some important data points that would be beneficial to discuss (if not in this paper due to space then in an extension of this paper). The four selected evaluation configurations (i.e., edge devices only, VFE, Conv1, Conv2) make sense but they would benefit from being compared to two additional configurations: (1) edge server only (with 3D point cloud transfer from edge device) and (2) edge device only but with a light-weight/optimized model (maybe LW-Sconv). While this paper is evaluating an SC approach, I believe that the edge server only model would provide a good reference point for better understanding the trade-offs being explored in the evaluation (i.e., inference time, data size, and data transfer time). Similarly, the “light-weight edge model only” configuration would also enable a discussion of model accuracy. I would expect the light-weight model to be less accurate but it would be beneficial to see how it compares to the other models with respect to accuracy and inference time. Furthermore, the addition of these two models would further bolster the privacy goals of this work as they reside on either end of the privacy spectrum (not protected <---> fully protected): the edge server configuration does not protect the point cloud data whereas “the light-weight edge device only” configuration does. Even if these additional configurations are not added, it is highly recommended to clarify (1) how these inference and data transfer times compare to the autonomous driving constraints (maybe add a dotted line or a shaded region in the plots) and (2) explicitly state that the model performance/accuracy does not change as a result of the proposed SC strategy (at least that seems to be the implication).

The authors mention privacy preservation as a main concern but they do not offer any citations or empirical evidence to support the threat of extracting private data from transmitted point cloud data. I do not doubt this is a legitimate concern but it would be useful to include some citations to sources describing such attacks in more detail in order to justify points made in the paper. For example, in Section IV, the final paragraph describes that point cloud data contains privacy information and that this privacy information can even be obtained from voxel data which is why the VFE configuration is not selected as the best option despite it having the lowest inference time.

Minor Notes/Comments/Questions:

I would be interested to know how potential network congestion is taken into account in an SC approach. Once again, safety-critical systems such as those in autonomous driving have tight timing constraints so an effective system would take into account the expected delay between sending data to the server and receiving the final result. Space permitting, a brief discussion around this would be interesting.

I would like to know what future work or remaining challenges the authors would consider as a follow up study to the study presented in this paper.

Is the Jetson Orin Nano indicative of the type of edge devices utilized in autonomous vehicles or relevant deployment areas? There are commercial SoC platforms such as the NVIDIA DRIVE product line [1] which specifically target autonomous vehicles. There are also FPGA-based solutions from companies such as AMD/Xilinx [2] that are targeting these tasks. While FPGAs may not have the capacity to store a full DNN model, there are many examples of deploying optimized DNN models onto an FPGA in order to achieve faster inference times [3a-3b].

Another interesting point of exploration for a larger work would be to include another evaluation configuration which performs some sort of transformation (e.g., encryption, compression) on the point cloud data prior to transferring it to the edge server. Particle physicists have used autoencoders [4] to compress large amounts of data (~40TB/s) from the Large Hadron Collider at CERN (note that this autoencoder has also been deployed onto FPGA using hls4ml). I imagine a similar strategy could be used here (possibly in conjunction with SC) to address the privacy concerns with transmitting point cloud or voxel data to the edge server. Note: this comment is merely meant to be informative - I understand this encryption/autoencoder strategy is not necessarily in scope for work aiming to do split computing.

References:

[1] https://www.nvidia.com/en-us/self-driving-cars/hardware/#:~:text=truth%20data%20collection.-,NVIDIA%20DRIVE%20Hyperion,digital%20clusters%2C%20and%20AI%20cockpits.

[2] https://www.amd.com/en/solutions/automotive.html

[3a] hls4ml: https://fastmachinelearning.org/hls4ml/reference.html

[3b] https://arxiv.org/abs/2205.07690

[4] <https://arxiv.org/pdf/2105.01683.pdf>

2: (受け入れる)

概要

本稿では、自律走行などのアプリケーションで見られるエッジデバイス上の3D点群検出のためのスプリットコンピューティング（SC）アプローチを提案する。提案するSC戦略は、(1)処理時間の短縮、(2)エッジデバイスの計算負荷の軽減、(3)3次元点群データのプライバシーの保護を目的としている。これを実現するために、著者のSC戦略は、既存のモデル（Voxel R-CNN）をヘッドモデルとテールモデルに分割し、それぞれエッジデバイスとエッジサーバーに配置して推論を実行する。異なるポイントでのモデル分割の効果について議論し、実験的評価の中で探求している。

長所

この論文は非常によく書かれており、わかりやすい。さらに、提案されているSCアプローチは明確に定義されており、彼らのターゲット・アプリケーションや他の既存のSC研究の文脈の中でうまく位置づけられている。著者らがシステムに対して設定した3つの目標（私の要約と論文のセクションIで述べられている）を考慮すると、提示された実験的評価は、他のモデル構成と比較して彼らのアプローチの「成功」を決定するために必要なメトリクス（すなわち、推論時間、データサイズ、データ転送時間）を健全に探求している。この投稿のための限られたスペースを考えると、著者らは彼らのアプローチの説明と評価において良い仕事をした。

弱点：

この論文は自律走行技術に重点を置いているが、そのようなシナリオにおけるエッジデバイスの3D点群検出のタイミング制約を全く提供していない。これらのタイミング制約に関連するいくつかの値を含めることで、評価で示された結果を文脈化することができます（例えば、すべてのモデル構成が自律走行システムの推論速度要件を満たしているかなど）。

評価はよくできているが、（スペースの関係で本稿に掲載しないとしても、本稿の延長で）議論した方が有益であろう重要なデータポイントがいくつかあった。選択された4つの評価構成（すなわち、エッジデバイスのみ、VFE、Conv1、Conv2）は理にかなっているが、2つの追加構成（（1）エッジサーバのみ（エッジデバイスからの3D点群転送を伴う）、（2）エッジデバイスのみだが軽量/最適化モデル（おそらくLW-Sconv）を伴う）と比較されることで恩恵を受けるだろう。この論文ではSCアプローチを評価していますが、エッジサーバーのみのモデルは、評価で検討されているトレードオフ（推論時間、データサイズ、データ転送時間など）をよりよく理解するための良い参照点を提供すると思います。同様に、「軽量エッジモデルのみ」の構成でも、モデルの精度について議論することができます。軽量モデルは精度が低いと予想されますが、精度と推論時間に関して他のモデルと比較することは有益でしょう。さらに、これら2つのモデルを追加することで、プライバシースペクトルの両端（保護されない<--->完全に保護される）に存在するため、この研究のプライバシー目標をさらに強化することができます。エッジサーバー構成では点群データが保護されないのに対し、「軽量エッジデバイスのみ」構成では保護されます。これらの追加構成が追加されないとしても、(1)これらの推論とデータ転送時間が自律走行制約と比較してどうであるかを明確にすること（多分、プロットに点線か斜線領域を追加すること）、(2)提案されたSC戦略の結果としてモデル性能/精度が変化しないことを明示すること（少なくともそれが暗示であるように思われる）を強く推奨する。

著者は主な懸念事項としてプライバシーの保護に言及しているが、送信された点群データからプライベートデータを抽出する脅威を裏付ける引用や実証的な証拠を提示していない。私はこれが正当な懸念であることを疑わないが、論文で指摘された点を正当化するためには、そのような攻撃をより詳細に記述したソースの引用を含めることが有用であろう。例えば、セクションIVの最後の段落では、点群データにはプライバシー情報が含まれており、このプライバシー情報はボクセルデータから取得することも可能であると記述されています。

細かいメモ/コメント/質問：

SCアプローチにおいて、潜在的なネットワークの輻輳がどのように考慮されているのか知りたい。繰り返しになるが、自律走行のようなセーフティ・クリティカルなシステムには厳しいタイミング制約があるため、効果的なシステムでは、データをサーバーに送信してから最終結果を受信するまでに予想される遅延を考慮する必要がある。スペースが許す限り、この辺りの簡単な議論は興味深いだろう。

この論文で発表された研究のフォローアップ研究として、著者らがどのような将来の仕事や残された課題を検討するのか知りたい。

Jetson Orin Nanoは、自律走行車や関連する配備分野で利用されるエッジデバイスの種類を示しているのでしょうか？NVIDIA DRIVE製品ライン[1]のような商用SoCプラットフォームがあり、これらは特に自律走行車をターゲットにしている。また、AMD/Xilinx [2]のようなFPGAベースのソリューションもあり、これらのタスクをターゲットにしています。FPGAには完全なDNNモデルを格納する容量はないかもしれないが、最適化されたDNNモデルをFPGAに展開することで、推論時間の高速化を実現している例は多い[3a-3b]。

また、エッジサーバーに転送する前に、点群データに対して何らかの変換（暗号化、圧縮など）を実行する別の評価コンフィギュレーションを含めることも、より大規模な研究のための興味深いポイントだろう。素粒子物理学者は、CERNの大型ハドロン衝突型加速器からの大量のデータ（～40TB/s）を圧縮するために、オートエンコーダ[4]を使用している（このオートエンコーダは、hls4mlを使用してFPGAにもデプロイされていることに注意）。点群やボクセルデータをエッジサーバーに送信する際のプライバシーに関する懸念に対処するために、ここでも（おそらくSCと組み合わせて）同様の戦略を使用できると想像している。注：このコメントは、単に情報提供を意図したものです。この暗号化/オートエンコーダ戦略は、スプリット・コンピューティングを目的とした作業の範囲には必ずしも含まれないと理解しています。

-1: (weak reject)

The paper combines split computing (SC) and 3D object detection at the Edge to reduce processing time and power consumption.

Main comments:

- The novelty is not well justified. The combination of SC, object detection, and edge computing seems not new in the literature. See for example the following random paper: [Matsubara, Yoshitomo, and Marco Levorato. "Split computing for complex object detectors: Challenges and preliminary results." Proceedings of the 4th International Workshop on Embedded and Mobile Deep Learning. 2020.]

- Authors offer a limited comparison with related studies. The performance evaluation offered has some value, but it would be fair to do it with comparable SoA approaches and/or to further detail why the comparison here is necessary.

- The paper needs to be rewritten in several sections. Many paragraphs repeat the information already presented in other sections. See, e.g. the system model.

- The system model section does not really present a system model. It is more of a background or preliminaries section. This challenges reproducibility.

- The contributions declared are not fair. As indicated before the combination of SC, object detection, and edge computing is already in the literature. What is the specific contribution/target of this paper wrt to the SoA?

- Overall, the contribution - if exists - is more practical than scientific. Still, in terms of practical details, the paper is not complete. With the information available in the paper it is difficult to ensure repeatability.

- The approach section also requires a significant intervention. A novel approach is not described here, and again, a repetition of information already written in other sections of the paper is there.

- Privacy is mentioned as a benefit of the approach. This aspect is kind of hidden, not explicit in the abstract, but highlighted as a key feature in other parts of the paper. It would be good if this aspect is better clarified consistently throughout the paper.

-1：（弱い拒絶反応）

スプリットコンピューティング(SC)とエッジでの3Dオブジェクト検出を組み合わせ、処理時間と消費電力を削減した論文。

主なコメント

- 新規性が正当化されていない。SC、物体検出、エッジコンピューティングの組み合わせは、文献上新しいものではないようだ。例えば、以下のランダムな論文を参照： [Matsubara, Yoshitomo, and Marco Levorato. "Split computing for complex object detectors： 課題と予備的結果". Proceedings of the 4th International Workshop on Embedded and Mobile Deep Learning. 2020.]

- 著者は関連研究との限定的な比較を行っている。提供された性能評価には一定の価値があるが、同等のSoAアプローチと比較すること、および／または、ここでの比較がなぜ必要なのかをさらに詳細に説明することが公平であろう。

- この論文はいくつかのセクションを書き直す必要がある。多くのパラグラフは、他のセクションで既に提示された情報を繰り返している。例えば、システムモデル。

- システムモデルのセクションは、実際にはシステムモデルを提示していない。むしろ背景や前置きのセクションである。これは再現性に問題がある。

- 宣言された貢献は公平ではない。先に示したように、SC、物体検出、エッジコンピューティングの組み合わせはすでに文献にある。SoAに対するこの論文の具体的な貢献・目標は何ですか？

- 全体的な貢献は、もしあるとすれば、科学的というより実用的です。それでも、実用的な詳細という点では、この論文は完全ではない。論文で利用可能な情報では、再現性を確保するのは難しい。

- また、アプローチのセクションにはかなりの介入が必要である。斬新なアプローチはここには書かれておらず、また、論文の他のセクションですでに書かれている情報の繰り返しがそこにある。

- アプローチの利点としてプライバシーが挙げられている。この側面は、抄録では明示されていないが、論文の他の部分では重要な特徴として強調されている。この点が論文全体を通して一貫してより明確にされると良いだろう。