Distributed Fusion Using Attention Based Deep Distributed Learning with Multimodal Conformer

Peipei Wu, Jinzheng Zhao, Özkan Çaylı, Wenwu Wang
Centre for Vision, Speech and Signal Processing
University of Surrey
Surrey, UK
Email: {peipei.wu, j.zhao, o.cayli, w.wang}@surrey.ac.uk

Yang Liu
Meta
Seattle, USA
Email: yangliuai@meta.com

***Abstract*—Distributed fusion has gained attention for its potential in tasks like tracking across sensor networks. Advances in communication and computational technologies enable deploying deep learning at network edges. This paper proposes a novel attention-based distributed fusion approach comprising three components: Optimal Distributed Set-Theoretic Information Flooding (ODSIF), an Encoding System (ES), and a Crossmodal-Conformer (CMCM). ODSIF exchanges and balances local and received data weights, while ES converts sensor node information into Gaussian heat maps. These heat maps are processed by CMCM to generate distributed fusion results independently at each node, eliminating the need for data association, a key step in traditional methods. The attention mechanism further mitigates outliers' impact, ensuring robustness with less accurate sensor data. Simulations using Stone Soup, featuring three targets and seven sensors, demonstrate the proposed method's superiority over DSIF, Confidence-Informed DSIF (C-DSIF), and Conformer.**

I. Introduction

With the advances in computational and communication technologies, distributed fusion has garnered significant interest from both industry and academia, in a wide range of applications, including distributed tracking [1], e-health [2], environmental and traffic monitoring [3], and battlefield surveillance [4]. Among these, tracking multiple targets from various viewpoints using a distributed sensor network is a crucial technology underpinning many applications, which forms the application example discussed in this paper.

Several approaches have been proposed for distributed fusion by reducing differences in the information shared by local agents throughout the network. These can generally be classified according to the applied factors, such as consensus on estimates (CE) [5], [6], consensus on measurements (CM) [7], [8], and consensus on information (CI) [9], [10]. To achieve consensus, in general, the arithmetic average (AA) or geometric average (GA) fusion is applied to those factors which have been discussed in detail in [11]. However, arithmetic average fusion is prone to potential outliers in the information shared between networks, coupled with its limited capacity to manage such outliers [1]. In contrast, geometric fusion involves various challenges, particularly in defining the weights and covariance terms, which complicates its implementation [12]. In addition, consensus algorithms are expected to be applied to the same target with information shared across networks to mitigate the degradation of the results due to the potential fusion of information from different targets.

The conventional consensus-based algorithms often involve achieving global agreements, adding more assumptions, and/or pre-defining hyperparameters. Unlike conventional methods, we explore the potential of using deep learning to achieve consensus among the nodes. With more devices at the network edge capable of supporting deep learning-based models, the use of a deep learning-based fusion algorithm has become increasingly viable. In this paper, we explore the potential of using attention-based deep learning models, such as transformer [13] and conformer [14], for the fusion of distributed sensor data. The transformer [13] incorporates attention mechanisms, enabling it to process sequences of variable length, such as sentences, in natural language processing (NLP). Following its success in NLP, transformer-based architectures such as Vision Transformer (ViT) [15] and Conformer [14] have set new benchmarks in computer vision and audio signal processing, respectively. Moreover, the quest for the use of attention mechanisms beyond a single modality has led to the development of multimodal fusion approaches within a transformer-based framework, notably in audio-visual [16] and other multimodal contexts [17]. Given these successes, we are interested in studying whether the transformer-based architecture could be used to fuse information in distributed tracking, and whether achieving consensus is still crucial in distributed fusion.

In this paper, we introduce a novel attention-based distributed fusion method for distributed target tracking, aiming to improve tracking accuracy in the presence of unreliable sensor measurements. This novel approach, distributed crossmodal-conformer (D-CMCM), comprises three main components, namely, an optimal distributed set-theoretic information flooding (ODSIF) algorithm, a shared information encoding system (ES), and a crossmodal-conformer (CMCM). In this approach, an attention mechanism is integrated into distributed fusion, thus eliminating the need for data association in distributed fusion to match the target identities and the need to achieve consensus on shared information which may contain subtantial outliers. The organization of this paper is as follows. The proposed algorithm is described in Section II. Section III presents

numerical evaluations of our approach. Finally, Section IV concludes the paper and outlines potential directions for future research.

II. Proposed D-CMCM Method

The proposed D-CMCM consists of three main modules: ODSIF, ES, and CMCM. The module ODSIF shares the information from each local agent, such as the local point estimates and distances between local and global estimates, across the distributed sensor network through the dynamic connection topology. Local agents then use the ES module to convert received local information into 2D heat maps. Moreover, the distance information determines the weighting factor for fusion within the cross-modal conformer. The 2D Gaussian heat maps, comprising the local information map $I^L \in \mathbb{R}^{H\times W\times 1}$ and received information map $I^R \in \mathbb{R}^{H\times W\times 1}$, are processed by the CMCM to provide the distributed fusion results.

A. The ODSIF Algorithm

At time t, each node in the distributed network compiles a specific set of information for sharing, denoted for the s-th sensor as $\mathcal{I}_t^s = \{\hat{\mathcal{X}}_t^s, d_{t-1}^s\}$. Here, $\hat{\mathcal{X}}_t^s = \{\hat{x}_t^1, \ldots, \hat{x}_t^n\}$ represents the set of local estimates of the target states, where n signifies the target index. The term d_{t-1}^s indicates the minimum Euclidean distance between the set of local estimates $\hat{\mathcal{X}}_{t-1}^s$ and the set of fused information $\overline{\mathcal{X}}_{t-1}^s$ at the time $t-1$.

Since sensor nodes communicate only with their neighbors in the topology, the information-sharing process is run iteratively to reach non-directly connected nodes. At time t, during the initial iteration $i = 0$, the known information set for the s-th sensor is $\mathcal{O}_t^s(0) = \mathcal{I}_t^s$, referred to as the occupied information set. Sensor s also receives information from its neighbors, denoted as $\mathcal{R}_t^s(0) = \bigcup_{j\in\mathcal{N}_t^s} \mathcal{O}_t^j(0)$, where $\mathcal{N}_t^s$ is the set of neighbors of the s-th sensor at time t, and j indexes the neighboring sensors. For simplicity, the time term t is omitted in this section.

During each iteration, the occupied information set is updated as follows:

$$\mathcal{O}^s(i) = \mathcal{O}^s(i-1) \cup \mathcal{R}^s(i-1), \tag{1}$$

where the received information set $\mathcal{R}^s(i)$ is updated by:

$$\mathcal{R}^s(i-1) = \bigcup_{j\in\mathcal{N}^s} \{\mathcal{O}^j(i-1) \setminus \mathcal{O}^s(i-1)\}, \tag{2}$$

where $\mathcal{O}^j(i-1)\setminus\mathcal{O}^s(i-1)$ includes only the novel information from the received set $\mathcal{O}^j(i-1)$ not already present in $\mathcal{O}^s(i-1)$. To prevent indefinite iteration, a termination condition must be established, which stops the process when the specified criterion is met. An ideal termination condition is when the received information set $\mathcal{R}^s(i) = \emptyset$, indicating that the sensor s has acquired all possible information from the network. However, in extensive networks with numerous sensor nodes, achieving this ideal state may be computationally prohibitive. To address this, we introduce the metric $C^s(i) = \frac{c(i)}{a(i)}$, which measures the proportion of the network from which the sensor s has received information, allowing data fusion using only a subset of the network. Here, $c(i)$ is the number of sensors that shared information with s, and $a(i)$ is the total number of active sensors, which was counted during the sharing. If $C^s(i)$ exceeds a pre-defined threshold $\mathtt{C}$, the sharing process is terminated.

Upon receiving information from other nodes, we derive weights from the distance term d in each information set $\mathcal{I}$ from the known set $\mathcal{O}$, assigning higher weights to estimates with smaller d, thus improving the subsequent fusion step. For sensor s, let d_L represent d_{t-1}^s as the local distance, and d_R denote the average distance derived from all distance values $d_{t-1}^{n\neq s}$ in the received information set $\mathcal{O}^s$. The weights assigned to the local and received information, denoted by ω_L and ω_R, respectively, are computed as follows:

$$\omega_L = \frac{d_R}{d_L + d_R}, \quad \omega_R = \frac{d_L}{d_L + d_R}. \tag{3}$$

The pseude code of ODSIF is given in Algorithm 1.

Algorithm 1 ODSIF Algorithm

Require: t, $\{\hat{\mathcal{X}}_t^s, d_{t-1}^s\} \ldots \{\hat{\mathcal{X}}_t^j, d_{t-1}^j\}$, $\mathcal{N}_t^s$, $i = 0$, $\mathtt{C}$
if $i = 0$ **then**
  Initialize the occupied information set: $\mathcal{O}_t^s(0) \leftarrow \mathcal{I}_t^s$
  Receive information from neighbor sensor nodes: $\mathcal{R}_t^s(0) \leftarrow \bigcup_{j\in\mathcal{N}_t^s} \mathcal{O}_t^j(0)$
end if
while True **do**
  Update the occupied information set as in Eq. (1)
  Update the received information set as in Eq. (2)
  $i \leftarrow i + 1$
  if $i \leq 3$ **then**
    if $\mathcal{R}^s(i) = \emptyset$ **then**
      break
    end if
  else if $i \leq 7$ **then**
    Calculate the metric: $C^s(i) \leftarrow \frac{c(i)}{a(i)}$
    if $C^s(i) > \mathtt{C}$ **then**
      break
    end if
  else
    break
  end if
end while
Calculate the weight terms as in Eq. (3)

B. Encoding System

In ODSIF, the set of local point estimates and distance measures is shared by local sensors. The ES converts elements from these sets into two Gaussian heat maps, i.e. indicating the possible estimates of the target real state, to support learning of the proposed model. As shown in Fig. 1, the image on the left shows the local information map, while the image on the right shows the received information map. Instead of normalizing the values of each pixel in the heat map, the maximum likelihood value of overlapping distributions is recorded, preserving peaks.

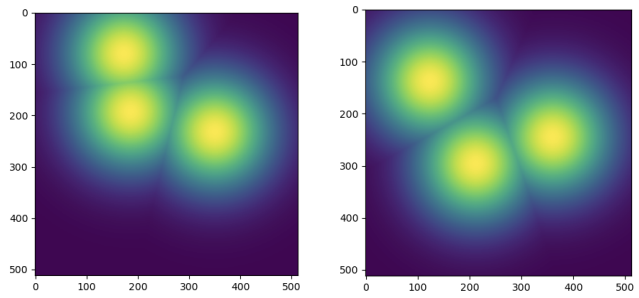

Fig. 1. Two Gaussian heat maps generated by the encoding system. The **left** is the local information map, and the **right** is the received information map.

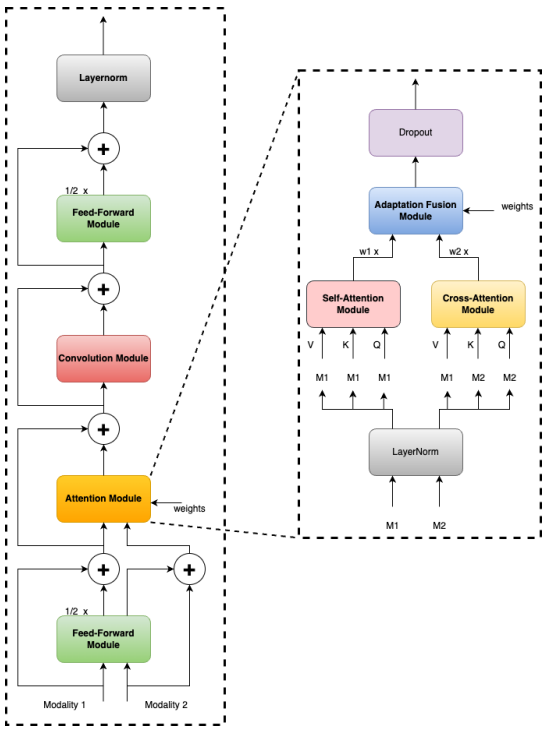

Fig. 2. The architecture of the Crossmodal-Conformer block, where different modalities are denoted as M1 and M2.

C. Crossmodal-Conformer Blocks

The CMCM component is designed to integrate received information with local data using attention mechanisms. Mirroring the traditional Conformer block structure [14], the CMCM block includes four sequential components: a feed-forward module, a novel attention module, a convolution module, and a final feed-forward module. While the final feed-forward and convolution modules are adopted from the original Conformer, the CMCM block introduces a new framework and replaces the standard self-attention module with a novel attention mechanism, as shown in Figure 2. This modification enhances the interaction between local and received information, optimizing information fusion within the distributed tracking framework.

1) Feed-Foward Module: The architecture of the feed-forward module in the CMCM block aligns with that of the Transformer and Conformer, as detailed in [13]. However, a notable adaptation in our approach is that the initial feed-forward module in the CMCM block is designed to process dual inputs, corresponding to the local information map and the received information map, from each respective modality. Additionally, we incorporate pre-normalization [18] and dropout techniques within this module to enhance its robustness and generalization capability. Before entering into the attention module, the residual module is applied with only half of the value from outputs of the first feed-forward module added to the residuals. The same operation is applied to the two modalities, respectively.

2) Attention Module: Distinct from the Transformer and Conformer, the attention module in the CMCM encompasses both self-attention and crossmodal attention mechanisms. The multi-headed self-attention (MHSA) mechanism [13] is traditionally applied within a single modality. In our CMCM's self-attention module, MHSA is employed to process the local information. To incorporate attention from the received information, distinct inputs are utilized, with Keys and Queries derived from the received information and Values from the local information, facilitating effective crossmodal attention integration.

The adaptation fusion (AF) module is introduced to integrate the outputs from the self-attention and cross-attention modules, denoted as SA and CA, respectively. Utilizing the weights derived from ODSIF, as specified in Equation (3), the output of the AF module is formulated as a weighted sum of the two attention module embeddings:

$$AF = \omega_L \cdot SA + \omega_R \cdot CA, \tag{4}$$

where the two weight terms ω_L and ω_R are assigned to the self-attention and cross-attention outputs following equation 3, respectively. A dropout module is also connected behind the adapt fusion module in the Attention Module.

III. Experimental Results

A. Experimental Settings

To address the lack of real-world datasets for distributed tracking, we used the Stonesoup Python package [19], [20] from DSTL, UK [1]. In a 512 × 512 square-meter environment, three targets are monitored by seven sensors with dynamic network topology, and trajectories generation using Extended Kalman Filter (EKF). Sensor estimates follow Gaussian distributions (std. dev. 80) to simulate unreliability. The experiment spans 1000 frames across four sequences, generating 7000 Gaussian heat maps for system testing in noisy conditions. A pretrained ResNet-18 [21] is used as feature extractor, followed by two CMCM blocks, each with four heads in both self-attention and cross-attention modules. The predictor only produces point estimates that exceed the confidence threshold. The learning rate (Adam optimizer) is set at 0.0005, reduced by 90% every 50 epochs, and the attention weights are set at 0.5 during training. During experiments, we simulate scenarios

[1]DSTL: Defence Science and Technology Laboratory, UK

where information from one, two, three, or all seven nodes is available to each local sensor for fusion. Evaluation metrics include OSPA [22], precision (error distance below a strict 20-pixel threshold), and success rate (ratio of successful steps across the entire trajectories).

B. Results on Simulations

TABLE I
OSPA COMPARISONS ON DIFFERENT APPROACHES.

Method	OSPA			
	1 Node	2 Node	3 Node	7 Node
DSIF-AA (pm)	79.9	40.3	39.5	35.4
DSIF-AA	79.9	68.7	50.3	48.6
C-DSIF-AA (pm)	79.9	39.2	36.7	32.5
C-DSIF-AA	79.9	50.2	48.8	41.5
ODSIF-Conformer	19.2	12.1	11.3	8.8
ODSIF-D-CMCM	**18.5**	**4.7**	**4.6**	**4.1**

TABLE II
PRECISION COMPARISONS ON DIFFERENT APPROACHES.

Method	Precision			
	1 Node	2 Node	3 Node	7 Node
ODSIF-Conformer	66.0	80.6	83.3	85.8
ODSIF-D-CMCM	**67.1**	**95.0**	**95.1**	**95.1**

TABLE III
ABLATION STUDY ON DIFFERENT CONFIGURATIONS.

Method	1 Node			2 Node		
	OSPA	PR	SR	OSPA	PR	SR
FE	20.2	61.3	58.2	20.0	61.8	58.6
FE+Conformer	19.2	66.0	64.4	12.1	80.6	78.7
FE+D-CMCM	**18.5**	**67.1**	**64.5**	4.8	94.9	**94.5**
FE+D-CMCM+W	**18.5**	**67.1**	**64.5**	**4.7**	**95.0**	**94.5**

To evaluate the effectiveness of our method, we compare it with the Conformer model [14], DSIF-AA [23] and C-DSIF-AA [1]. Comparisons are conducted under two consensus conditions: partial consensus (information from a subset of the network) and complete consensus (information from the entire network). For a comprehensive evaluation, we employ two matching settings: perfect matching (correct target associations) and random matching (arbitrary associations). ODSIF collects information from other nodes to feed both Conformer and D-CMCM. However, Conformer only accepts a single input, and ODSIF does not provide weight outputs for Conformer, unlike D-CMCM. Additionally, the encoding system compresses local and received information into the same map for Conformer.

1) OSPA Comparision: Table I shows OSPA metrics for different methods across configurations. ODSIF-D-CMCM achieves the lowest OSPA distances, indicating superior fused tracking accuracy. The performance gap between Conformer and D-CMCM narrows with a single sensor node due to reduced cross-attention impact, but both outperform AA fusion methods. With more nodes, distributed fusion enhances performance for all methods. However, AA methods require good data association, such as perfect matching, to obtain this improvement. Unlike AA methods, deep learning approaches can avoid this requirement. Notably, D-CMCM maintains consistent accuracy across 2, 3, and 7 nodes, demonstrating resilience to network consensus variations and suitability for diverse scenarios.

2) Precision Comparision: To evaluate the performance of deep learning-based methods in distributed fusion, we employ a custom-defined precision metric. As illustrated in Table II, the precision of both ODSIF-Conformer and ODSIF-D-CMCM is assessed in various configurations involving a different number of participating nodes. Consistent with the observed OSPA performance trends, an increase in the number of participating nodes correlates with an improvement in precision for both methods. In particular, even with information sourced from only a portion of the network, both methods achieve a high level of precision, indicating that complete network consensus is not critical for deep learning-based distributed fusions to maintain high performance. Furthermore, the comparison reveals a consistent performance improvement of ODSIF-D-CMCM over ODSIF-Conformer under identical settings, underlying the beneficial impact of the cross-attention mechanism in enhancing the effectiveness of distributed fusion.

C. Ablation Studies

Ablation studies were conducted to evaluate the contributions of individual components in our deep learning-based distributed fusion models, as detailed in Table III. The results show significant performance gains with deep learning-based fusion methods, even with two sensor nodes, prompted by the focus on one- and two-node configurations. Although adding cross-attention to self-attention offers limited benefits with sparse external information, its impact becomes substantial in two-node setups where received information is utilized. Additionally, a weighting mechanism that balances self- and cross-attention further enhances performance.

IV. CONCLUSION

In this study, we departed from traditional consensus-based approaches to distributed fusion by incorporating attention mechanisms to enhance local fusion processes with insights from received information. We have introduced D-CMCM, a novel deep learning-based distributed fusion method that combines ODSIF, an encoding system, and a Crossmodal-Conformer. The ODSIF algorithm facilitates dissemination of information and calibrates the influence of local versus received data for fusion. The encoding system converts information into a heat map format for integration into the deep learning framework, while the Crossmodal-Conformer refines local estimates using the attention mechanism. Our extensive numerical experiments validate D-CMCM's effectiveness and superiority, potentially setting new benchmarks in distributed fusion. Future work will focus on using temporal sequences in tracking tasks and addressing challenges posed by limited sensing fields, and further enhancing distributed fusion system capabilities.

REFERENCES

[1] P. Wu, J. Zhao, S. Goudarzi, and W. Wang, "Partial Arithmetic Consensus based Distributed Intensity Particle Flow SMC-PHD Filter for Multi-Target Tracking," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 5078–5082.

[2] E. Fadel, V. C. Gungor, L. Nassef, N. Akkari, M. A. Malik, S. Almasri, and I. F. Akyildiz, "A survey on wireless sensor networks for smart grid," *Computer Communications*, vol. 71, pp. 22–33, 2015.

[3] B. E. Bilgin and V. C. Gungor, "Adaptive error control in wireless sensor networks under harsh smart grid environments," *Sensor Review*, vol. 32, no. 3, pp. 203–211, 2012.

[4] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[5] P. Millán, L. Orihuela, C. Vivas, and F. R. Rubio, "Distributed consensus-based estimation considering network induced delays and dropouts," *Automatica*, vol. 48, no. 10, pp. 2726–2729, 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0005109812003639

[6] S. Zhu, C. Chen, X. Ma, B. Yang, and X. Guan, "Consensus Based Estimation Over Relay Assisted Sensor Networks for Situation Monitoring," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 2, pp. 278–291, 2015.

[7] P. J. Legree, J. Psotka, T. Tremble, and D. R. Bourne, "Using consensus based measurement to assess emotional intelligence," *Emotional Intelligence: An International Handbook*, pp. 155–179, 2005.

[8] S. Robertson, P. Kremer, B. Aisbett, J. Tran, and E. Cerin, "Consensus on measurement properties and feasibility of performance tests for the exercise and sport sciences: a Delphi study," *Sports Medicine-Open*, vol. 3, no. 1, pp. 1–10, 2017.

[9] G. Battistelli, L. Chisci, G. Mugnai, A. Farina, and A. Graziano, "Consensus-Based Linear and Nonlinear Filtering," *IEEE Transactions on Automatic Control*, vol. 60, no. 5, pp. 1410–1415, 2015.

[10] ——, "Consensus-based algorithms for distributed filtering," in *Proceedings of 51st IEEE Conference on Decision and Control (CDC)*. IEEE, 2012, pp. 794–799.

[11] K. Da, T. Li, Y. Zhu, H. Fan, and Q. Fu, "Recent advances in multisensor multitarget tracking using random finite set," *Frontiers of Information Technology & Electronic Engineering*, vol. 22, no. 1, pp. 5–24, 2021.

[12] T. Li, V. Elvira, H. Fan, and J. M. Corchado, "Local-diffusion-based distributed SMC-PHD filtering using sensors with limited sensing range," *IEEE Sensors Journal*, vol. 19, no. 4, pp. 1580–1589, 2018.

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[14] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100*, 2020.

[15] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[16] Y.-B. Lin and Y.-C. F. Wang, "Audiovisual transformer with instance attention for audio-visual event localization," in *Proceedings of the Asian Conference on Computer Vision*, 2020.

[17] Y.-H. H. Tsai, S. Bai, P. P. Liang, J. Z. Kolter, L.-P. Morency, and R. Salakhutdinov, "Multimodal transformer for unaligned multimodal language sequences," in *Proceedings of the conference. Association for Computational Linguistics. Meeting*, vol. 2019. NIH Public Access, 2019, p. 6558.

[18] Q. Wang, B. Li, T. Xiao, J. Zhu, C. Li, D. F. Wong, and L. S. Chao, "Learning deep transformer models for machine translation," *arXiv preprint arXiv:1906.01787*, 2019.

[19] P. A. Thomas, J. Barr, B. Balaji, and K. White, "An open source framework for tracking and state estimation ('Stone Soup')," in *Proceedings of Defense + Security*, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:67294683

[20] J. Barr, O. Harrald, S. Hiscocks, N. Perree, H. Pritchett, S. Vidal, J. Wright, P. Carniglia, E. Hunter, D. Kirkland, D. Raval, S. Zheng, A. Young, B. Balaji, S. Maskell, M. Hernandez, and L. Vladimirov, "Stone Soup open source framework for tracking and state estimation: enhancements and applications," in *Proceedings of Signal Processing, Sensor/Information Fusion, and Target Recognition XXXI*, I. Kadar, E. P. Blasch, and L. L. Grewe, Eds., vol. 12122, International Society for Optics and Photonics. SPIE, 2022, p. 1212205. [Online]. Available: https://doi.org/10.1117/12.2618495

[21] S. Targ, D. Almeida, and K. Lyman, "Resnet in resnet: Generalizing residual architectures," 2016. [Online]. Available: https://arxiv.org/abs/1603.08029

[22] B. Ristic, B.-N. Vo, D. Clark, and B.-T. Vo, "A metric for performance evaluation of multi-target tracking algorithms," *IEEE Transactions on Signal Processing*, vol. 59, no. 7, pp. 3452–3457, 2011.

[23] T. Li and F. Hlawatsch, "A distributed particle-PHD filter using arithmetic-average fusion of Gaussian mixture parameters," *Information Fusion*, vol. 73, pp. 111–124, 2021.