文章编号：2097-1974(2025)04-0053-06 DOI：10.7654/j.issn.2097-1974.20250407

面向高超声速飞行器的高效智能计算系统

徐勇军，刘杭达，安梓嘉，刁博宇
（中国科学院计算技术研究所，北京，100190）

摘要：以美国黑燕高超声速飞机的实时遥感分析为例，分析典型智能化高超声速飞行器对强实时、高能效智能计算的需求。在此基础上，分别从智能模型轻量化、软硬件协同编译优化和超异构融合计算硬件三个层次详细分析如何构建面向智能化高超声速飞行器的强实时、高能效智能计算系统，并给出系统集成示例。未来通过强实时高能效智能计算系统的加持，智能化高超声速飞行器将更具自主性、可靠性与群体协同能力，可推动空天跨域飞行变革和全球快速穿梭领域的智能化升级。

关键词：高超声速飞行器；智能化；强实时；高能效；智能计算系统
中图分类号：V27 **文献标识码**：A

High-efficiency Intelligent Computing System for Hypersonic Aircraft

XU Yongjun, LIU Hangda, AN Zijia, DIAO Boyu
(Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100190)

Abstract: This article takes the real-time intelligent reconnaissance of the American "Black Swift" hypersonic intelligent aircraft as an example to analyze the demand for strong real-time, high-energy efficiency intelligent computing of typical intelligent hypersonic vehicles. On this basis, it analyzes in detail how to build a strong real-time, high-energy efficiency intelligent computing system for intelligent hypersonic vehicles from three levels: intelligent model lightweight, software and hardware collaborative compilation and optimization, and ultra-heterogeneous integrated computing hardware. Furthermore, a system integration example is provided to illustrate the practical application of these principles. In the future, with the loading of the strong real-time, high-energy efficiency intelligent computing system, intelligent hypersonic aircraft will be more autonomous, reliable and capable of group collaboration, and will drive the intelligent upgrade of aerospace cross-domain flight and global rapid transit.

Keywords: hypersonic vehicles; intelligent; hard real-time; high energy efficiency; intelligent computing system

0 引 言

智能化高超声速飞行器已成为大国战略和先进空天技术的新热点。高超声速飞行器在跨域飞行和全球快速穿梭上大有可为，能够在超高速空天环境下实现自主感知、自主认知、任务协同等复杂功能。这些自主功能的实现主要依赖搭载于飞行器上的智能计算系统，由于高超声速飞行器一般具有高机动、小型化的特点，智能计算系统必须具备强实时和高能效的特点，以满足高超声速飞行器的智能计算需求。

1 高超声速飞行器分类及技术需求

高超声速飞行器是指能够在大气层内外以5马赫（约6 125 km/h）以上的速度飞行的飞行器。这些飞行器能够在相对较短的时间内飞越大陆或海洋，具有显著的快速机动和随遇处置能力。高超声速飞行器的研发是现代航空技术的前沿领域，其发展对未来商业航空、全球快速穿梭具有重要意义。

高超声速飞行器早期更多服务于军事领域，近几年在民用航空领域获得飞速发展，且在商业上大获成功。通常分为以下3类：

a）高超声速滑翔飞行器（Hypersonic Glide Vehicles，HGV）。这类飞行器通常由火箭发射到大气层边缘，然后以高超声速滑翔至目标区域。通常用于远程机动遥感分析，并且由于高速和飞行路径的不可预测，这类飞行器成本高且难以重复使用，商业化还需探索。

b）高超声速飞机（Hypersonic Aircraft，HA）。这类飞行器旨在以高超声速运送人员或货物，它们可

收稿日期：2024-10-21；修回日期：2025-07-02

以用于商业航空，以大大减少长途飞行的时间，或者用于遥感分析和控制任务，其突出优势是可重复使用。

c）空天飞机或可重复使用的大推力火箭。这类空天飞机能以类似速度飞行，并且能够直接在常规跑道起飞和着陆，进入和返回太空，成本低、机动灵活，特别是大推力火箭，在近几年获得较大发展，是未来商业航空航天的重要发展方向。

近年来，人工智能技术在各个领域的研究上被广泛应用[1]，而其中在高超声速飞行器上的应用已成为近几年大国技术博弈的新热点。在推进、材料和热防护、气动设计以及导航和控制等传统关键技术之外，智能计算技术也成为了飞行器研制的重要组成。

高超声速飞行器的智能计算系统需要面对机动灵活、可靠性高、任务多样、资源严重受限等苛刻要求，技术需求主要分为以下几个方面。

a）实时信号处理和控制：高超声速飞行器的飞控系统需要对飞行器进行实时控制和数字信号处理，包括对内部状态进行监测、测试、诊断等操作，实时处理对信息处理机的瞬时性能要求极高。

b）智能态势感知和理解：态势感知和理解能力是高超声速飞行器设计的关键，需要具备强实时、全空间感知能力，随着深度学习算法的应用，智能化空间态势感知，为飞行器的大规模商业化发展提供可能。

c）多飞行器群智化工作：智能化高超声速飞行器需要具备群体协同工作能力，以完成多飞行器的空中接力，通过高速数据链实现信息共享和任务规划，可提高空间多主体群体发展和综合工作效能。

d）自主规划和决策能力：高超声速飞行器远离地面，必须具备自主推理、自主判断和自主决策能力，能够在远离管控中心/人员、超高速飞行的条件下自主完成复杂任务。

小型化与低功耗设计是这类智能系统的主要方向，需要在有限的空间和能耗下实现高性能计算，以适应未来各种空天飞行器的体积、质量、能源限制。为了满足实时性和低功耗的要求，这类计算平台往往需要采用超异构智能计算架构，以加速多样化智能计算任务；此外，还需要在各种未知、不确定、恶劣环境下稳定工作，要求其具有高可靠性和抗干扰能力。

2 智能化高超声速飞行器

目前民用领域的高超声速飞行器的发展主要是以空天遥感和航空运载为主，例如高超声速飞机，结合了高超声速飞行和人工智能技术，是全球最先进技术的集大成者。本节以美国黑燕高超声速飞机为例，介绍智能化高超声速飞行器的技术现状。

美国黑燕高超声速飞机，也被称为SR-72，是美国最新前沿项目之一，旨在开发一种速度超过5马赫的高性能跨域飞行器，可实现大范围实时遥感影像获取和分析。主要具备以下技术特点：

超高速飞行：SR-72的设计目标是达到6马赫（约6 437 km/h）的速度，这将使其成为有史以来最快的飞机之一。这样的速度能够显著缩短从美国到达全球任何地点的时间。

高海拔飞行：SR-72的设计飞行高度约24~26 km，这比当前任何商用飞机都要高，使其能够在绝大多数防空系统的射程之外实现自主飞行。

SR-72的主要职能是大范围高机动遥感影像分析，这对机载智能计算提出了极为严格的要求，需要具备高分辨率、高帧率的遥感影像实时分析能力。例如对30 帧/s 10K×10K分辨率的遥感影像进行实时分析，需要智能计算系统具备74 TOPS以上的峰值算力输出，而由于机载环境功耗和体积严格受限，智能计算载荷的额定功率一般不会超过30 W，即计算系统的算力能效比需超过2.5 TOPS/W，现有智能计算系统在面对不同任务类型时的推理性能差异明显，满足SR-72的实时遥感分析需求依然面临挑战。

3 强实时高能效智能计算系统

智能计算系统是高超声速飞行器在复杂环境下执行任务的核心组成，须确保在有限空间内的高效计算，以保障任务的顺利执行。因此需要设计一种面向实时性和能效优化的强实时、高能效智能计算系统，旨在通过模型轻量化、软硬件协同编译优化和超异构计算的融合，提升计算性能和资源利用效率。

3.1 智能模型的轻量化技术

智能模型的轻量化技术旨在通过探索神经网络中的过度参数化和结构冗余，并将其移除，以获得更高效的模型结构。通过减少模型的计算需求和存储占用，轻量化技术能降低能耗并加快推理速度，并通过增量学习的方式保证模型精度[2-3]。该技术在高超声速飞行器等对计算性能和能效有严格要求的应用场景中显得尤为重要。典型的轻量化方法包括网络剪枝和

低比特量化[4]。这类方法为在飞行器上部署强实时高能效的智能计算系统提供了关键支持，满足严苛条件下对模型高效性和实时性的需求。

3.1.1　神经网络剪枝和优化技术

目前，可用于神经网络剪枝的方法通常根据剪枝的粒度分为两类：非结构化剪枝和结构化剪枝。非结构化剪枝在最细粒度的水平上进行剪枝，即权重级剪枝，遵循以下优化问题：

$$\begin{aligned}&\min_{\theta} L(\theta; D)\\&\text{s.t.}\left\|\theta\right\|_0 \leqslant k\end{aligned} \tag{1}$$

式中　L为数据集D上的通用损失函数；θ为模型参数；k为目标非零权重数。虽然非结构化剪枝通常可以极大地减少参数大小或内存占用，但它不能保证推理延迟的加速，因为剪枝后的模型形状往往不规则，需要配合特定的硬件设计。

相比之下，将被删除的结构指定为整个层、头或其他网络单元的剪枝称为结构化剪枝。给定一个特定的剪枝比例和一个神经网络$S=\{s_1,s_2,\cdots,s_L\}$，其中s_i可以是第i层中的通道、滤波器、神经元或Transformer的注意力头。结构化剪枝的目标是寻找$S'=\{s_1',s_2',\cdots,s_L'\}$，以在给定的剪枝比例下，最大程度地减少性能下降并最大化提升速度，其中$s_i' \subseteq s_i, i \in \{1,2,\cdots,L\}$。由于这种剪枝方法剪枝后的模型形状形式与原模型一致，因此通常可以在标准硬件上缩短推理耗时，从而减少对高超声速飞行器硬件平台的额外设计需求。网络剪枝已经被广泛用于不同类型的深度学习模型中，表1展示了不同深度学习模型代表性剪枝方法的效果。

表1　不同深度学习模型代表性剪枝方法

Tab.1　Representative pruning methods for different deep learning models

方法	任务	模型	参数量	FLOPs/G	加速比	精度/%
GFP[5]	分类	ResNet[6]	19.4 M/25.6 M	4.1/2.0	1.79	−0.37
		MobileNet-V2[7]	3.50 M/3.3 M	0.29/0.30	1.25	+0.5
	检测	RetinaNet[8]	26.3 M/37.9 M	119/239	1.57	0.0
SAViT[9]	分类	DeiT-B[10]	25.4 M/86.6 M	5.3/17.6	2.05	−0.1
	检测	Faster R-CNN[11]	19.2 M/45.2 M	68/222	—	−0.3
Sparse GPT[12]	语言建模	OPT[13]	1.35 B/2.7 B	—	1.79	−0.39

3.1.2　低比特量化技术

量化是将深度学习模型部署到各种设备（尤其是具有专门用于低精度算术的电路的GPU和NPU）时的重要步骤。在量化过程中，浮点张量x会被转换为整数张量x_{int}，并记录相应的量化参数（缩放因子s和零点z）。根据保存的量化参数，整数张量x_{int}可以反量化回浮点数x_{quant}，过程如下式所示[4]：

$$\begin{aligned}x_{\text{int}} &= \text{Clamp}\left(\left\lfloor \underline{x}/\overline{s} \right\rceil + z, 0, 2^b - 1\right)\\ x_{\text{quant}} &= s\left(x_{\text{int}} - z\right)\end{aligned} \tag{2}$$

式中　b为位宽；$\lfloor\cdot\rceil$表示对数值（用•表示）作四舍五入；Clamp用于截断超出给定范围的值。虽然可以通过反量化得到浮点数x_{quant}，但其与原始的浮点张量x相比会产生一定的精度误差，导致模型性能的下降。

在模型推理过程中，嵌入张量e和权重w量化为定点数后，配合保存的量化参数可以将二者的浮点乘法转换为定点乘法，过程如下所示：

$$\begin{aligned}y = \text{MatMul}(e,w) &\approx \text{MatMul}\left(e_{\text{quant}}, w_{\text{quant}}\right) =\\ &\text{MatMul}\left(s_{\text{e}}\left(e_{\text{int}} - z_{\text{e}}\right), s_{\text{w}} w_{\text{int}}\right) =\\ &s_{\text{e}} s_{\text{w}} \text{MatMul}\left(e_{\text{int}}, w_{\text{int}}\right) + C\end{aligned} \tag{3}$$

式中　s_{e}、z_{e}为嵌入张量的量化参数；s_{w}为权重的量化参数，由于权重的分布通常是对称的，因此往往采用对称量化，简化为零点$z_{\text{w}}=0$；常数C可以通过s_{e}、z_{e}、s_{w}和w_{int}预先计算；e_{int}和w_{int}是整数形式的输入和权重。通过这种方式，模型推理过程中的浮点数乘法可以转换为高效的整数乘法，以适用于高超声速飞行器平台搭载的小型化低功耗硬件平台。低比特量化技术作为模型部署的关键，用于各类深度学习模型，表2展示了低比特量化各类深度学习模型的效果。

表2　不同深度学习模型代表性量化方法

Tab.2　Representative quantization methods for different deep learning models

方法	任务	模型	量化位数	精度/%
PD-Quant[14]	分类	ResNet[6]	W4/A4	−1.47
			W2/A2	−19.47
FQN[15]	检测	Retinanet[8]	W4/A4	−3.1
Quantformer[16]	分类	DeiT-S[10]	W4/A4	−1.7
			W2/A2	−14.7
	检测	MaskRCNN[17]	W4/A4	−1.3
			W2/A2	−4.7
BiViT[18]	分类	ViT[19]	W1/A1	−12.25
SmoothQuant[20]	完形填空	OPT[13]	W8/A8	−0.1

3.2 软硬件协同的编译优化技术

近年来，硬件架构呈现出多样化和复杂化的趋势，而手工优化技术由于需要大量的专业知识和开发时间，使其难以快速支持新算子和新硬件，难以满足实际应用需求。现代编译优化框架通过自动化调优技术，结合硬件特性和算子特征，通过反复组合、测试并比较不同的优化方案，来发现最佳优化策略，并在硬件平台上生成高效的执行代码。这种软硬件协同的编译优化技术减少了手动开发成本，甚至可以发现人类专家忽视的优化策略。本文研究了几种主流的网络基础算子编译优化技术，包括TVM、Ansor和Roller等，并探讨它们在不同模型中的应用和性能提升。这些方法作为关键技术有效增强了飞行器上智能计算系统的强实时性和高能效性。

TVM、Ansor和Roller是目前学术界和工业界广泛使用的三种编译优化工具。它们将神经网络的网络基础算子抽象为张量程序中间表示（tensor Intermediate Representation，tensor IR），通过尝试包含循环分块（tile）和多级缓存（cache）等技术的不同优化操作组合，寻找具有更高并行性的算子执行代码，从而实现更优的性能。

TVM是一个端到端的深度学习编译器[21]，支持从模型定义到硬件执行的全栈优化。TVM通过定义一个广泛的调度搜索空间，并利用基于代价模型的优化策略，能够在不同硬件平台上自动生成高效执行代码；其核心在于抽象硬件特性（如缓存级数、缓存大小和峰值算力等）和算子需求，自动实现算子的最优调度和数据布局。TVM的调优过程通常包括算子分块、线程分配和数据重排等多个步骤，极大地提升了推理性能。

Ansor是TVM的扩展工具[22]，专注于通过随机采样等技术加速自动化调优过程。Ansor引入了一种基于机器学习的成本模型，通过快速迭代和高效搜索，能够在短时间内找到最优调度方案。Ansor的优势在于能够在硬件资源有限的情况下处理更复杂的搜索空间。式（4）描述了Ansor成本模型：

$$L(f,P,y)=W_{\mathrm{p}}\left(\sum_{s\in S(P)}f(s)-y\right)^2 \tag{4}$$

式中 f为成本模型；P为算子程序；y为算子程序的吞吐量；W_{p}为程序权重，与程序的吞吐量成正比；$S(P)$代表算子程序中的非循环计算代码。

以上两种工具均为搜索式的编译优化技术，虽然通过自动化方法提升了算子性能，但是它们的缺点在于在复杂的搜索空间中搜索会花费大量时间。Roller[23]提出了一种完全不同的基于树结构的构造式编译优化技术，避免复杂的搜索过程，从而减少优化时间。它专注于探索与底层硬件处理器单元（Processing Units）特性对齐的张量形状，并采用预定义的规则和递归算法来直接构建基于分块的高效算子程序。Roller无需搜索即可完成编译优化，将算子编译优化的速度由小时级提高到秒级。然而，Roller存在构建程序性能不佳的问题，原因在于Roller采用了树结构的优化空间，而树结构的单方向性和目标单一性容易导致程序陷入局部最优解[24]。本文提出基于图结构的张量编译优化方法Graph-Roller，扩展构建空间，支持优化状态回溯，从而有效满足在强实时高超声速飞行的环境下深度学习模型优化的需求。

表3列出了在应用软硬件协同编译优化技术后，几个常用深度学习模型在推理性能上的显著提升效果。具体来说，该表展示了模型在未优化和优化后的推理时间，以及相应的速度提升倍数。这些优化方法通过充分利用硬件特性和算子特性，显著提高模型的推理效率。

表3 软硬件协同编译优化对推理性能的提升效果

Tab.3 Impact of hardware-software co-optimization on inference performance

模型类型	优化方法	未优化推理时间	优化后推理时间	速度提升倍数
YOLO-V3[2]（目标检测模型）	TVM[21]	40.21 ms	14.46 ms	2.78
ResNet[6]（图像识别模型）	Ansor[22]	2496 ms	478 ms	5.22
Bert[25]（预训练模型）	Roller[23]	8955 ms	2 539 ms	3.53
GPT-2[26]（大语言模型）	Roller	116.98 s	44.03 s	2.66
DeepSeekR1-1.5B[27]（大语言模型）	Graph-Roller	353.28 ms	298.22 ms	1.18

在实际应用中，高超声速飞行器需要在极短时间内处理大量多模态传感器数据（如雷达、红外、导航等），并实时完成目标识别、轨迹规划、环境监控等复杂任务。这些场景对计算效率、延迟和可靠性提出了极高要求。通过软硬件协同编译优化技术，本文针对飞行器嵌入式平台（GPU、NPU、FPGA等）对深度学习模型进行了性能优化和能效提升。首先利用张量编译技术，加速了模型对大规模传感器数据的处理能力。此外，张量编译的快速优化功能支持动态生成优化代码，适应飞行器运行时动态变化的计算需求，

进一步增强了模型的实时性与适应性。在高超声速飞行器的模拟任务测试中，优化后的模型在目标识别任务中延迟降低至毫秒级，使用大模型进行轨迹规划等决策任务的计算速度提升18%以上，同时内存带宽压力显著降低，为飞行器的智能化决策与控制提供了强有力的技术支持。

3.3　超异构融合的系统优化技术

超异构融合计算是一种新兴的计算架构，通过集成多种不同类型的处理器（如CPU、GPU、NPU、FPGA、DSA、ASIC等），来实现更高能效的计算[28]。在超异构融合智能计算系统中，软硬件协同优化是关键，这不仅仅是在物理层面上将不同的处理器集成到一个系统中，更重要的是在软件层面上实现深度协同和融合。这意味着，系统需要跨不同类型的处理器运行计算任务，同时保持高性能和灵活性。为了实现这一点，需要开发统一的软件运行环境和编译开发工具，以降低系统的复杂度，并实现计算任务的跨平台运行。

此外，超异构计算系统通常采用数据流驱动的模式，而不是传统的指令流驱动。这是因为在超异构系统中，由于系统复杂度的影响，指令流的设计模式很难同步不同系统间的控制交互。因此，整个系统的运转可以理解为由数据流驱动。

3.4　超异构融合计算架构的系统集成

本文设计了一个高效的计算架构，如图1所示，通过CPU居中调度，充分发挥各类计算单元（NPU、GPU、FPGA）的独特优势，从而实现性能提升、速度加快与功耗降低的综合目标。CPU作为系统的核心调度单元，负责任务分配、资源管理与多智能体协同，确保各计算单元高效协作。NPU（如BM1684X）专注于图像识别任务，利用其高效的并行计算能力处理视觉数据，同时通过轻量化模型（如剪枝、量化）进一步优化计算效率。GPU负责大模型推理任务，包括多模态轨迹规划等，通过张量编译优化（如算子融合、内存优化）和硬件感知优化（如缓存优化、指令集优化），显著提升推理速度与算力能效比。FPGA负责物理场反演（通过观测数据推断和重建未知物理场分布），系统使用孪生计算模型构建实时物理场，并对其进行针对性加速优化。在系统层面，统一的编译优化框架将轻量化模型与超异构计算紧密结合，实现任务的无缝分配与资源的高效利用，并将系统整体功率控制在25 W以下，算力能效比在4 TOPS/W以上。

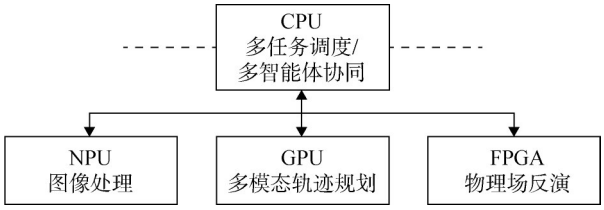

图1　超异构融合系统结构

Fig.1　Schematic of ultra-heterogeneous integrated system

在高超声速飞行器计算任务中，CPU根据任务复杂度动态分配计算负载，NPU处理图像输入，承担视觉功能，GPU使用大语言模型辅助完成多模态轨迹规划，承担决策功能，FPGA使用孪生计算模型进行物理场反演，承担环境监控功能，同时系统可以通过分布式优化与流水线处理技术隐藏通信与计算延迟。表4列出了系统在多任务推理中的性能表现。实际应用中，这种超异构融合的系统优化技术不仅显著提升了计算性能与实时性，还通过功耗优化与资源复用降低了整体能耗，为智能化高超声速飞行器提供了高效、灵活的计算解决方案。

表4　超异构融合硬件平台在多任务推理中的性能表现

Tab.4　Performance evaluation of multi-task inference on ultra-heterogeneous integrated hardware platforms

处理器平台	模型	推理速度	准确率
NPU 图像识别-视觉数据集-imagenet-val	ResNet-50[6]	462.96 帧/s	A_{cc}=80.00%
	YOLO-V5[2]	83.89 帧/s	IoU=0.53
GPU 大语言模型辅助轨迹规划-决策数据集-arc_challenge	DeepSeekR1-1.5B[27]	29.62 tokens/s	A_{cc} =0.35
	Qwen2.5-0.5B[29]	54.19 tokens/s	A_{cc} =0.29
FPGA 物理场反演-环境监控数据集-TFR-HSS	U-Net[30]	2.784 帧/s	MAE=3.03
	SegNet[31]	1.420 帧/s	MAE=3.98

在实际应用中，超异构融合智能计算系统可以提供接近于ASIC的极致性能，同时保持接近于CPU软件的通用可编程能力。这种系统的发展对于应用于高超声速飞行器的强实时、高能效智能计算系统来说是至关重要的，其优势体现在以下5个方面：

a）提升性能：通过集成不同类型的处理器，如

CPU、GPU、DSP、FPGA等，可以充分发挥各种计算单元的优势，实现计算性能的最大化。例如，在多媒体处理、大数据分析、人工智能等领域，异构SoC能够提供强大的并行处理能力，加速复杂计算任务的执行。

b）降低功耗：异构计算系统可以根据任务需求合理分配计算资源，通过发掘软件算法中的并行性，降低功耗，实现超低功耗计算。例如，根据负载需求的不同，可以将任务分配给功耗较低的处理器执行，从而减少能源消耗。

c）软硬件融合：软硬件融合的超异构计算通过系统级的协同设计，打破软硬件的界限，实现整体最优[32]。这种融合不仅提升了性能，还使得硬件更加灵活，功能更加强大，从而更好地适应特定场景的计算需求。

d）解决特定领域问题：异构SoC特别适合解决特定领域的复杂计算问题，如自动驾驶、5G/6G核心网、边缘计算等。这些场景对算力的需求极高，异构SoC通过集成多种处理器引擎，提供了所需的计算能力和灵活性。

e）适应复杂工作负载：随着数据量的爆炸式增长和工作负载的多样化，传统的单一计算架构已经无法满足需求。异构SoC通过集成多种计算资源，能够更好地处理多类型任务，适应复杂的工作负载。

4 结束语

智能化高超声速飞行器正成为未来商业航空航天领域的重要发展方向，其结合了高超声速飞行技术与人工智能技术，为跨域飞行和全球快速穿梭提供了全新的解决方案。然而，实现智能化高超声速飞行器需要强实时、高能效的智能计算系统的支持，为此需要通过模型轻量化、软硬件协同编译优化和超异构计算融合等方式，集成为超异构融合计算系统，联合提升计算性能和资源利用效率。

致 谢

感谢中国科学院稳定支持基础研究领域青年团队项目（YSBR-107）对本研究的支持，感谢中科华创（杭州）科技有限公司为本研究提供算力资源支持。

参 考 文 献

[1] XU Y, LIU X, CAO X, et al. Artificial intelligence: a powerful paradigm for scientific research[J]. The Innovation, 2021(2). DOI: 10.1016/j.xinn.2021.100179.

[2] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: unified, real-time object detection[C]. Las Vegas: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[3] AN Z, DIAO B, HUANG L, et al. IOR: inversed objects replay for incremental object detection[C]. Hyderabad: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2025.

[4] HAN S, MAO H, DALLY W J. Deep compression: compressing deep neural network with pruning, trained quantization and huffman coding[C]. San Juan: International Conference on Learning Representations (ICLR), 2016.

[5] LIU L, ZHANG S, KUANG Z, et al. Group fisher pruning for practical network compression[C/OL]. International Conference on Machine Learning (PMLR), 2021.

[6] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]. Las Vegas: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[7] SANDLER M, HOWARD A, ZHU M, et al. MobilenetV2: inverted residuals and linear bottlenecks[C]. Salt Lake City: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[8] ROSS T Y, DOLLÁR G. Focal loss for dense object detection[C]. Hawaii: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[9] ZHENG C, ZHANG K, YANG Z, et al. Savit: structure-aware vision transformer pruning via collaborative optimization[J]. Advances in Neural Information Processing Systems, 2022(35): 9010-9023.

[10] TOUVRON H, CORD M, DOUZE M, et al. Training data-efficient image transformers & distillation through attention[EB/OL]. (2022-12-14)[2024-10-10]. https://arxiv.org/abs/2212.07048.

[11] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 39 (6): 1137-1149.

[12] FRANTAR E, ALISTARH D. Sparsegpt: massive language models can be accurately pruned in one-shot[EB/OL]. (2023-11-02) [2024-10-10]. https://arxiv.org/abs/ 2301.00774v1.

[13] ZHANG S, ROLLER S, GOYAL N, et al. Opt: open pre-trained transformer language models[EB/OL]. (2022-06-21) [2024-10-10]. https://arxiv.org/abs/2205.01068.

[14] LIU J, NIU L, YUAN Z, et al. Pd-quant: post-training quantization based on prediction difference metric[EB/OL]. (2022-12-14) [2024-10-10]. https://arxiv.org/abs/2212.07048.

[15] LI R, WANG Y, LIANG F, et al. Fully quantized network for object detection[C]. Long Beach: 2019 Proceedings of the IEEE/CVF Conference on Computer Vision and Ppattern Recognition, 2019.

[16] WANG Z, WANG C, XU X, et al. Quantformer: learning extremely low-precision vision transformers[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 45(7): 8813-8826.

（下转第80页）

空间碎片减缓详细要求》历程回顾与解读[J]. 空间碎片研究, 2021, 21(3): 34-38.
TANG Mingliang, GU Yanfeng, WANG Ying. Process review and interpretation of “ISO 20893: Space Systems——Detailed Space Debris Mitigation Requirements for Launch Vehicle Orbital Stages” [J]. Space Debris Research, 2021, 21(3):34-38.

[3] 胡锐锋, 吴子牛, 曲溪, 等. 空间碎片再入烧蚀预测与地面安全评估软件系统[J]. 航空学报, 2011, 32(3): 390-399.
HU Ruifeng, WU Ziniu, QU Xi, et al. Debris reentry and ablation prediction and ground risk assessment software system[J]. Acta Aeronaut ET Astronaut Sinica, 2011, 32(3): 390-399.

[4] REYNOLDS R C, SOTO A. Debris assessment software: operators manual[G]. NASA Johnson Space Center, 2001.

[5] FRITSCHE B, KLINKRAD H, KASHKOVSKY A, et al. Spacecraft disintegration during uncontrolled atmospheric re-entry[J]. Acta Astronaut, 2000, 47(2/9): 513-522

[6] PONTIJAS FUENTES I, BONETTI D, LETTERIO F, et al. Upgrade of ESA's debris risk assessment and mitigation analysis (DRAMA) tool: spacecraft entry survival analysis module[J]. Acta Astronautica, 2017, 158(5): 148-160.

[7] BONETTI D, FUENTES I P, LETTERIO F, et al. Upgrade of the spacecraft entry survival analysis module (SESAM) of the ESA's debris risk assessment and mitigation analysis (drama) tool[C]. Milan: 7th European Conference for Aeronautics and Space Sciences, 2017.

作 者 简 介

李洪波（1979—），女，博士，研究员，主要研究方向为飞行器总体设计、空间态势评估、航天战略与体系研究。
李宇飞（1978—），男，博士，高级工程师，主要研究方向为飞行器总体设计。
胡 超（1988—），男，高级工程师，主要研究方向为空间目标特性分析、空间态势感知。
席福彪（1984—），男，博士，高级工程师，主要研究方向为体系设计与评估、地理信息与空间态势分析。

（上接第58页）

[17] HE K, GKIOXARI G, PIOTR D, et al. Mask R-CNN[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017. DOI: 10.1109/TPAMI.2018.2844175.

[18] HE Y, LOU Z, ZHANG L, et al. Bivit: Extremely compressed binary vision transformers[EB/OL]. (2022-12-20)[2024-10-10]. https://arxiv.org/abs/2211.07091.

[19] DOSOVITSKIY A. An image is worth 16×16 words: transformers for image recognition at scale[EB/OL]. (2020-10-22) [2024-10-10]. https://arxiv.org/abs/2010.11929.

[20] XIAO G, LIN J, SEZNEC M, et al. Smoothquant: accurate and efficient post-training quantization for large language models[EB/OL]. (2022-11-18)[2024-10-10]. https://arxiv.org/abs/2211.10438.

[21] CHEN T, MOREAU T, JIANG Z, et al. TVM: an automated end-to-end optimizing compiler for deep learning[EB/OL]. (2018-12-12) [2024-10-10]. https://arxiv.org/abs/1802.04799.

[22] ZHENG L, JIA C, SUN M, et al. Ansor: generating high-performance tensor programs for deep learning[EB/OL]. (2020-06-11)[2024-10-10]. https://arxiv.org/abs/2006.06762.

[23] ZHU H, WU R, DIAO Y, et al. ROLLER: fast and efficient tensor compilation for deep learning[C] . Carlsbad: 16th USENIX Symposium on Operating Systems Design and Implementation (OSDI 22), 2022.

[24] LIU H, DIAO B, YANG Y, et al. Gensor: a graph-based construction tensor compilation method for deep learning[EB/OL]. (2025-02-17) [2025-03-03]. https://arxiv.org/abs/2502.11407.

[25] ALAPARTHI S, MISHRA M. Bidirectional encoder representations from transformers (BERT): a sentiment analysis odyssey[EB/OL]. (2020-07-02)[2024-10-10]. https://arxiv.org/abs/2007.01127.

[26] RADFORD A, WU J, CHILD R, et al. Language models are unsupervised multitask learners[J]. OpenAI Blog, 2019, 1(8): 9.

[27] LIU A, FENG B, XUE B, et al. DeepSeek-V3 technical report[EB/OL]. (2024-11-27)[2025-03-02]. https://arxiv.org/abs/2412.19437.

[28] DONG D, JIANG H, DIAO B. AKGF: automatic kernel generation for DNN on CPU-FPGA[J]. The Computer Journal, 2024, 67(5): 1619-1627.

[29] YANG A, YANG B, ZHANG B, et al. Qwen2.5 technical report[EB/OL]. (2025-01-13)[2025-02-25]. https://arxiv.org/abs/2412.15115.

[30] RONNEBERGER O, FISCHER P, BROX T. U-net: Convolutional networks for biomedical image segmentation[C]. Munich: Medical Image Computing and Computer-assisted Intervention—MICCAI, 2015.

[31] BADRINARAYANAN V, KENDALL A, CIPOLLA R. Segnet: a deep convolutional encoder-decoder architecture for image segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(12): 2481-2495.

[32] LIU H, DIAO B, CHEN W, et al. A resource-aware workload scheduling method for unbalanced GEMMs on GPUs[J]. The Computer Journal, 2025, 68(3): 273-282.

作 者 简 介

徐勇军（1979—），男，博士，研究员，主要研究方向为智能计算系统。
刘杭达（1998—），男，博士研究生，主要研究方向为智能计算系统。
安梓嘉（1998—），男，博士研究生，主要研究方向为智能计算系统。
刁博宇（1989—），男，博士，高级工程师，主要研究方向为智能计算系统。