边缘协作的轻量级安全区域建议网络

熊金波[1,2]，毕仁万[1,2]，陈前昕[1,2]，刘西蒙[3,4]

（1. 福建师范大学数学与信息学院，福建 福州 350117；2. 福建师范大学福建省网络安全与密码技术重点实验室，福建 福州 350007；3. 福州大学数学与计算机科学学院，福建 福州 350108；4. 福州大学网络系统信息安全福建省高校重点实验室，福建 福州 350108）

摘 要：针对边缘环境下的图像隐私泄露和计算效率问题，提出一种边缘协作的轻量级安全区域建议网络（SecRPN）。基于加性秘密共享方案设计一系列安全计算协议，由 2 台非共谋边缘服务器协作执行安全特征处理、安全锚变换、安全边界框修正、安全非极大值抑制等计算模块。理论分析证明了 SecRPN 的正确性和安全性，实际性能评估表明，计算和通信开销均远优于现有工作。

关键词：边缘协作；区域建议网络；目标检测；加性秘密共享；安全计算协议

中图分类号：TP309.2

文献标识码：A

doi: 10.11959/j.issn.1000−436x.2020186

Towards edge-collaborative, lightweight and secure region proposal network

XIONG Jinbo[1,2], BI Renwan[1,2], CHEN Qianxin[1,2], LIU Ximeng[3,4]

1. College of Mathematics and Informatics, Fujian Normal University, Fuzhou 350117, China
2. Fujian Provincial Key Laboratory of Network Security and Cryptology, Fujian Normal University, Fuzhou 350007, China
3. College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350108, China
4. Fujian Provincial Key Laboratory of Network System Information Security, Fuzhou University, Fuzhou 350108, China

Abstract: Aiming at the problem of image privacy leakage and computing efficiency in edge environment, a lightweight and secure region proposal network (SecRPN) was proposed. A series of secure computing protocols were designed based on the additive secret sharing scheme. Two non-collusive edge servers cooperate to perform calculation modules such as secure feature processing, secure anchor transformation, secure bounding-box correction, and secure non-maximum suppression. Theoretical analysis guarantees the correctness and security of SecRPN. The actual performance evaluation shows that SecRPN is outstanding in the computational cost and communication overhead compared with the existing works.

Key words: edge-collaborative, region proposal network, object detection, additive secret sharing, secure computing protocol

1 引言

随着人工智能和物联网技术及其覆盖面不断扩延，智慧城市[1]、智能家居[2]、智能物流等概念自提出以来就受到广泛关注。根据中国智能物联网（AIoT, artificial intelligence and Internet of things）白

收稿日期：2020–05–06；**修回日期**：2020–07–20

通信作者：刘西蒙，snbnix@gmail.com

基金项目：国家自然科学基金资助项目（No.61872088, No.U1905211, No.U1804263, No.61702105, No.61872090）；福建省自然科学基金资助项目（No.2019J01276）；贵州省公共大数据重点实验室开放课题基金资助项目（No.2019BDKFJJ004）；陕西省网络与系统安全重点实验室开放课题基金资助项目（No.NSSOF1900104）；广东省数据安全与隐私保护重点实验室开放课题基金资助项目（No.2017B03031004-12）

Foundation Items: The National Natural Science Foundation of China (No.61872088, No.U1905211, No.U1804263, No.61702105, No.61872090), The Natural Science Foundation of Fujian Proince (No.2019J01276), The Guizhou Provincial Key Laboratory of Public Big Data Research Fund (No.2019BDKFJJ004), The Opening Project of Shaanxi Key Laboratory of Network and System Security (No.NSSOF1900104), The Opening Project of Guangdong Provincial Key Laboratory of Data Security and Privacy Protection (No.2017B03031004-12)

皮书预测[3]，到 2025 年，中国物联网设备接入量将达 200 亿台。自动驾驶车辆[4]、视频监控[5]、机器人等依赖物联网设备的群智感知视觉应用[6]，同时扮演着数据生产者和消费者的角色，通过装载的高清摄像头实时拍摄室内或室外场景信息，正确识别视野内目标的类别和位置，进而提供适当的行为策略或应用服务。

考虑目标检测任务本身的复杂开销问题，智能设备通常将预处理后的图像数据及检测任务外包给第三方进行存储和分析处理[7-8]，以最大程度地减少存储空间占用、计算开销和设备电源损耗。同时，由于网络容量、带宽等限制，边缘计算范式[9]将计算任务卸载至网络边缘，可以大大降低智能终端与边缘节点间的通信时延，从而满足时延敏感的智能应用需求。

目前，已经有学者结合边缘计算展开目标检测模型研究[10-11]，Ren 等[12]针对实时监控的应用需求，利用边缘计算来实现分布式、时延敏感的目标检测任务，有效地降低了通信开销和应用程序部署成本。Nikouei 等[13]提出了一种轻量级卷积神经网络，并将该网络部署在边缘节点上，利用边缘计算的优势实现了实时的密集行人目标检测。Zhang 等[14]提出了一种匹配边缘计算设备与建议区域的滤波算法，该算法利用有限的计算能力和内存来实现目标的跟踪和检测，并能保持较高的精度和较低的计算开销。

就目标检测技术而言，根据对检测精度和计算开销之间的权衡程度，主流的目标检测模型可以分为单目标检测与双目标检测模型两类[11]。Girshick 等[15]最早提出了一种分类和位置检测分离的目标检测方法 R-CNN（region-convolutional neural network），利用选择性搜索方式裁剪获得固定数量的目标位置区域，将这些区域缩放为固定尺寸后，顺序地通过卷积神经网络（CNN, convolutional neural network）进行分类[16]。显然，这需要大量的时间开销，缩放过程中的裁剪操作也会破坏完整的图像信息，导致检测精度下降。为了高效、准确地捕捉目标区域，Ren 等[17]提出了优化的端对端网络模型 Faster R-CNN，利用卷积神经网络获取图像的特征图，并引入了区域建议网络（RPN, region proposal network）概念，利用多种长宽比和尺寸的锚在特征图上进行检测，根据每个目标的残差值生成对应的边界区域，通过共享图像特征和边界框坐标，同时获得所有目标的位置及其类别。相比于双目检测模型，Redmon 等[18]将目标检测视为包含类别信息的位置回归问题，提出端对端的单目检测模型（YOLO, you only look once），具有计算效率方面的优势，但这类模型的检测精度相对较低。

然而，摄像头收集的图像数据通常包含大量隐私信息，例如自动驾驶车辆所拍摄的图像涉及特定目标的位置和车辆自身的移动轨迹信息[5]、室内监控图像包含私人的生活环境信息[6]等。当数据拥有者将原始数据与任务提交至边缘节点时，边缘节点是否可信、边缘环境是否存在恶意敌手是数据拥有者无从得知的，这意味着原始数据所包含的隐私信息存在泄露的风险，因此，原始数据必须以密文形式上传至边缘节点进行存储或处理[19]。目前，学者们主要基于同态加密（HE, homomorphic encryption）[20]或加性秘密共享方案来寻求处理密态数据的解决方法。Dowlin 等[21]提出了基于 HE 的隐私保护卷积神经网络模型 CryptoNets，该模型允许数据所有者以加密形式将数据发送至云服务器来执行图像分类任务，但只能支持简单的线性运算。为了弥补网络模型单一的缺陷，Hesamifard 等[22]设计了一些低阶多项式函数，利用线性 HE 可以近似处理线性修正单元（ReLU, rectified linear unit）、Sigmoid 函数等激活操作。Juvekar 等[23]结合 HE 和安全两方计算，设计了一些支持向量的密态计算协议。相比于基于 HE 的解决方案，在牺牲少量通信开销的前提下，Huang 等[24]结合加性秘密共享方案设计了安全乘法与安全比较协议，提出了一种轻量级特征提取框架，具有较高的执行效率。然而，ReLU 激活层耗费了大量时间开销计算符号进位加法，难以应用于实时深层 CNN 任务。Liu 等[25]提出了第一种隐私保护目标检测模型 SecRCNN，设计的一系列安全计算协议可以保证检测过程中图像数据的隐私性，但多轮迭代计算的特征使目标检测任务需要耗费大量的时间开销。

针对当前目标检测模型只专注于检测精度和执行效率的提升与优化，忽视了图像数据的隐私性，本文借鉴数据信息全生命周期的隐私保护和计算体系结构[26]，基于加性秘密共享方案为 RPN 设计相应的安全模型，同时贴合网联自动驾驶车辆等实时应用的低延时需求，在不影响图像隐私性的前提下尽可能地压缩时间和通信开销。本文的主要贡献具体如下。

1) 基于加性秘密共享方案设计了安全激活

（SRU, secure ReLU）、安全 Softmax（SST, secure softmax）、安全锚变换（SAT, secure anchor transform）、安全边界框裁剪（SBC, secure bounding-box clip）、安全边界框过滤（SBF, secure bounding-box filter）、安全非极大值抑制（SNMS, secure non-maximum suppression）等安全计算协议，实现网络函数功能的同时避免泄露隐私信息。相比于同态加密原语和多轮迭代逼近方法，本文方案具有计算和通信复杂度优势。

2) 提出一种安全 RPN 结构 SecRPN，智能终端随机地拆分图像数据并分别上传至边缘节点，2 台边缘服务器调用上述安全计算协议协同执行 SecRPN，依次包含安全特征处理、安全锚变换、安全边界框修正、安全非极大值抑制等计算模块，最终获得图像内目标的边界框和相应的分类概率。由于双方均不能获得完整的计算结果，SecRPN 可以保证目标位置区域和所属类别的隐私性。

3) 通过完备的理论分析证明安全计算协议和 SecRPN 的正确性、安全性和高效性。实验性能评估表明，SecRPN 的计算误差可以维持在 10^{-5} 左右，所获得的安全边界框与明文环境下的安全边界框几乎完全重合，并且时间成本仅为 0.34 s。

2 基础知识

2.1 区域建议网络

RPN 将图像内拟合的目标边界区域提取出来，分别送入全连接层模块进行分类，进而实现目标位置和类别检测的双重目的。在提取 CNN 特征图后，RPN 分别利用大小为 2×9 和 4×9 的单位卷积核执行逐点卷积操作，起到联通所有特征通道的作用。围绕每个特征点匹配 9 种不同比例和尺寸的锚（示范边界框），根据锚的左上角和右下角坐标，以及是否包含目标的分数进行描述，并利用 Softmax 函数将分数映射至区间[0,1]。为了获得所需要的目标边界框，RPN 将锚 $\{x_{\text{up}},x_{\text{bottom}},y_{\text{up}},y_{\text{bottom}}\}$ 变换为 $\{w_a,h_a,x_a,y_a\}$，其中，$(x_{\text{up}},y_{\text{up}})$ 和 $(x_{\text{bottom}},y_{\text{bottom}})$ 分别表示锚的左上角和右下角坐标，w_a、h_a 和 (x_a,y_a) 分别表示锚的宽、高和中心坐标，然后利用三类方向上的位移值 $(\mathrm{d}w,\mathrm{d}h,\mathrm{d}x,\mathrm{d}y)$ 执行锚变换获得目标边界框 $\{w_b,h_b,x_b,y_b\}$，即计算 $w_b=w_a\mathrm{e}^{\mathrm{d}w}$、$h_b=h_a\mathrm{e}^{\mathrm{d}h}$、$x_b=w\mathrm{d}x+x_a$ 和 $y_b=h_a\mathrm{d}y+y_a$。随后，执行的裁剪和过滤操作的目的是为了将边界框控制在图像边界内，并删除小于单位面积的无意义边界框。NMS 协议用于剔除检测分数较低、冗余的边界框，保留分数较高、有价值的边界。为了判定 2 个边界框的相似程度，本文定义交占比（IoU, intersection over union）为边界框重叠区域面积与覆盖区域面积的比值，当 IoU 低于设定的阈值时，则认为 2 个边界框是相似的，进而保留检测分数较高的检测框。经过这一系列操作之后，所得到的一批目标边界框可以代表整个特征图内的目标区域。

2.2 基本安全计算协议

相比于复杂的同态加密运算，加性秘密共享方案通过将输入值 x 随机地拆分为 2 份加法副本 x_1 和 $x_2(x=x_1+x_2)$，并分别发送给参与方 $\mathcal{S}_1$ 和 $\mathcal{S}_2$；$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 计算 $f_1(x_1)$ 和 $f_2(x_2)$ 这个过程等同于计算 $f(x)$，其中，f_1、f_2 和 f 表示函数。本文基于加性秘密共享方案设计安全协议，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 通过传递均匀分布的随机值进行交互。此外，本文采用固定点数完成数据存储和计算，任意数值可表示为 $u=(-1)^{\text{bit}}\bar{u}10^{-\text{dec}}$ 格式，其中，bit 和 dec 分别表示数值的符号和小数位数。将 u 与 $10^{-\text{dec}}$ 相乘得到整数 $\bar{u}$，随机选择 $\bar{u}_1\in\mathbb{Z}_n$，n 为足够大的素数，可以得到另一副本 $\bar{u}_2=\bar{u}-\bar{u}_1$。如果没有特殊说明，后文内容省略横线符号。在之前的工作中[24, 27]，基于加性秘密共享方案提出了 3 种基本的安全计算协议，具体构造过程参考附录，描述如下。

安全乘法（SMul, secure multiplication）协议[24]：$\mathcal{S}_1$ 拥有 $u_1,v_1\in\mathbb{Z}_n$，$\mathcal{S}_2$ 拥有 $u_2,v_2\in\mathbb{Z}_n$，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同计算 $f_1,f_2\leftarrow\text{SMul}(u_1,u_2,v_1,v_2)$，满足 $f_1+f_2=(u_1+u_2)(v_1+v_2)$。

安全指数（SExp, secure exponent）协议[27]：$\mathcal{S}_1$ 拥有 $u_1'\in\mathbb{Z}_n$，$\mathcal{S}_2$ 拥有 $u_2'\in\mathbb{Z}_n$，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同计算 $f_1',f_2'\leftarrow\text{SExp}(u_1',u_2')$，满足 $f_1'+f_2'=\mathrm{e}^{u_1'+u_2'}$。

安全比较（SComp, secure comparison）协议[27]：$\mathcal{S}_1$ 拥有 $u_1'',v_1''\in\mathbb{Z}_n$，$\mathcal{S}_2$ 拥有 $u_2'',v_2''\in\mathbb{Z}_n$，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同计算符号位 $f''\leftarrow\text{SComp}(u_1'',u_2'',v_1'',v_2'')$，若 $(u_1''+u_2'')\geqslant(v_1''+v_2'')$，则 $f''=0$；若 $(u_1''+u_2'')<(v_1''+v_2'')$，则 $f''=1$。

3 模型定义

3.1 系统模型

本文旨在解决智能物联网环境下数据采集设备将目标检测任务移动至边缘节点导致的图像数

据隐私问题，并尽可能地减少计算和通信开销浪费。系统模型如图 1 所示，参与实体包含数据拥有者$\mathcal{O}$、可信第三方服务器$\mathcal{T}$、2 台边缘服务器($\mathcal{S}_1$和$\mathcal{S}_2$)、应用服务提供商$\mathcal{P}$，具体职能描述如下。

1) $\mathcal{O}$ 负责采集物联网智能终端中的实时图像，将特征图随机拆分为 2 份加法副本，并分别提交给$\mathcal{S}_1$和$\mathcal{S}_2$。

2) $\mathcal{T}$ 仅负责离线生成随机数，并按照协议要求分别传递给$\mathcal{S}_1$和$\mathcal{S}_2$。

3) 接收到上传的加法特征副本后，$\mathcal{S}_1$和$\mathcal{S}_2$将加法副本隐藏至随机数中进行交互，根据安全计算协议执行安全特征处理、安全锚变换、安全边界框修正、安全 NMS 等模块操作，然后将各自获得的检测结果副本发送给$\mathcal{P}$。

4) $\mathcal{P}$ 仅需要执行加法便可以恢复出完整的检测结果，包含目标的类别和位置信息。

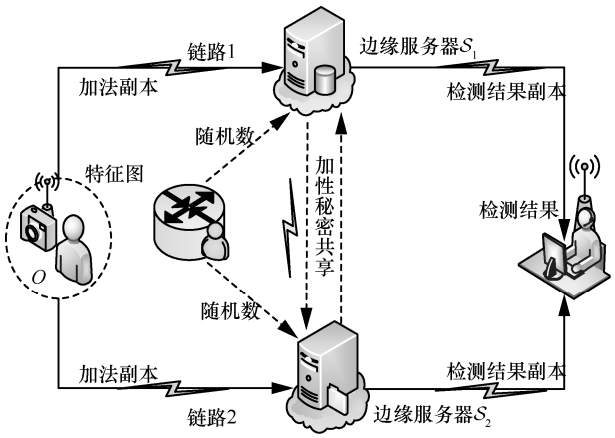


图 1　SecRON 网络模型

3.2　安全模型

本文定义隐私为检测目标的类别和位置信息，具体表现为网络处理过程中检测目标的每一位特征值。在半可信模型中，$\mathcal{T}$ 对于其他实体而言是完全可信的，不直接参与$\mathcal{S}_1$和$\mathcal{S}_2$的交互计算，因此不会影响模型的安全性。$\mathcal{S}_1$和$\mathcal{S}_2$被认为是诚实且好奇的实体，严格遵循所设计协议的要求，并期望通过已知信息推测出完整的隐私信息。$\mathcal{S}_1$和$\mathcal{S}_2$是不能共谋的，只能通过传递随机数进行交互，并且安全模型参与方的信息传递需要经过安全信道，以避免信息被恶意篡改。在计算过程中，若完整的隐私信息不会被$\mathcal{S}_1$和$\mathcal{S}_2$及概率多项式时间敌手$\mathcal{A}$截获，那么认为提出的模型是安全的。

类似于安全模型定义[24-25,28-30]，假设具备以下攻击能力：①$\mathcal{A}$至多可以窃听一类（图 1 中的“链路 1”或“链路 2”）通信链路并获得传递的特征副本；② $\mathcal{A}$至多可以破坏一台边缘服务器（$\mathcal{S}_1$或$\mathcal{S}_2$）并获得拥有的特征副本；③$\mathcal{A}$不能恶意干扰数据拥有者与$(\mathcal{S}_1,\mathcal{S}_2)$、$\mathcal{T}$与$(\mathcal{S}_1,\mathcal{S}_2)$、$\mathcal{P}$与$(\mathcal{S}_1,\mathcal{S}_2)$、$\mathcal{S}_1$与$\mathcal{S}_2$之间的正常通信，不能篡改传递的信息内容。

4　构造 SecRPN

4.1　SecRPN 概述

SecRPN 结构如图 2 所示。当接收到 2 份特征图副本后，$\mathcal{S}_1$和$\mathcal{S}_2$顺序地交互执行安全特征处理、安全锚变换、安全边界框修正和安全 NMS 等模块操作，然后分别输出目标检测结果副本。为了便于区分，$\mathcal{S}_1$执行图 2 中深灰色部分的操作，$\mathcal{S}_2$执行浅灰色部分的操作。在安全特征处理模块中，$\mathcal{S}_1$和$\mathcal{S}_2$利用大小为 3×3 的卷积核计算线性卷积，并在 ReLU 激活层中将 2 份加法副本之和的负特征值设置为 0。为了联通所有深度特征通道，$\mathcal{S}_1$和$\mathcal{S}_2$利用大小为 1×1 的卷积核协同执行逐点卷积操作生成目标边界框的分数和位移特征，额外地，$\mathcal{S}_1$和$\mathcal{S}_2$需要执行安全 Softmax 操作将锚的分数映射至区间[0, 1]。随后，$\mathcal{S}_1$和$\mathcal{S}_2$生成 9 种不同比例和尺寸的锚，并利用目标边界框的位移特征执行安全的锚变换操作。安全边界框修正操作的目的是将逾越图像边界的边界框限制在图像边界内，并且剔除小于单位大小的边界框。根据边界框的概率大小及修正边界框之间的重叠程度，$\mathcal{S}_1$和$\mathcal{S}_2$执行安全 NMS 操作删除概率较低的相似边界框。

4.2　安全特征处理模块

特征图中隐含着图像目标的类别和位置边界信息，经过安全特征处理操作，$\mathcal{S}_1$和$\mathcal{S}_2$可以采用 2 路卷积模块输出目标的分类概率和边界框坐标位移量。关于常规卷积和逐点卷积的线性计算，$\mathcal{S}_1$和$\mathcal{S}_2$分别拥有特征图副本x_1和x_2，已知公共的卷积核权重和偏置参数(ω,b)，$\mathcal{S}_1$和$\mathcal{S}_2$利用加性秘密共享独立计算$z_1=\omega x_1+b$和$z_2=\omega x_2+b$，可以获得完整特征图(x_1+x_2)的卷积结果，即$z_1+z_2=\omega x+b$。然而，ReLU 激活层负责计算非线性函数$\max(y,0)$，这显然不能直接拆分，因此本文提出了 SRU 协议。已知输入z_1和z_2，满足$z=z_1+z_2$，$\mathcal{S}_1$和$\mathcal{S}_2$协同执行 SComp 协议获得y与 0 的比较结果，即z的符号

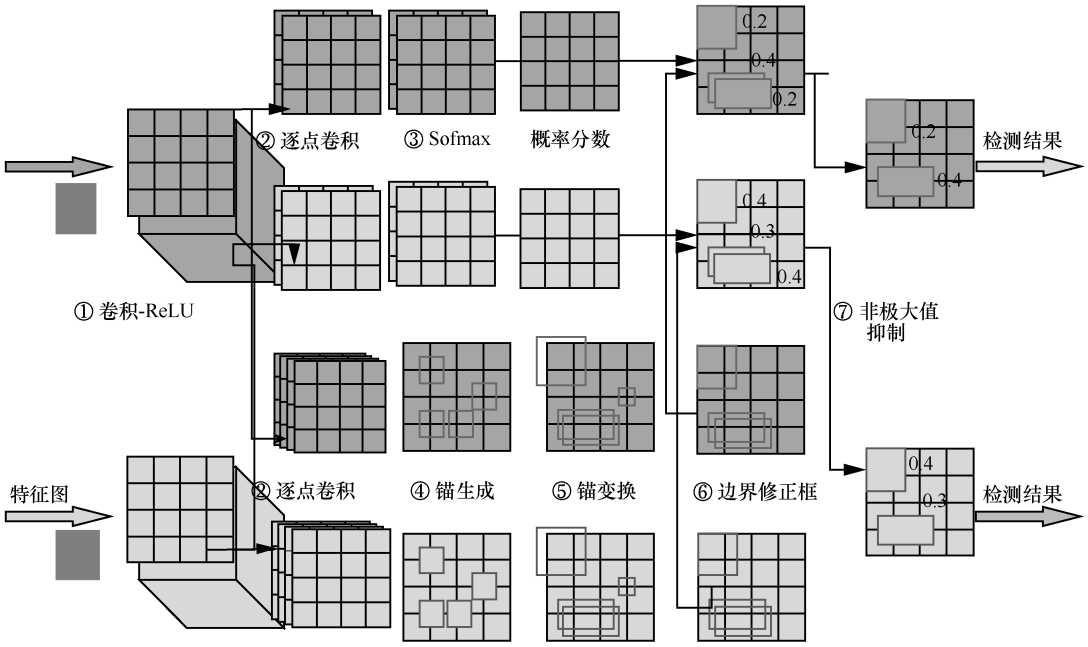


图 2　SecRPN 结构

位 b。若 $z \geqslant 0$，则 $b=0$；若 $z<0$，则 $b=1$。$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 直接将输入与 $\hat{b}$ 相乘获得激活结果，若 $z \geqslant 0$，则 $z_1'+z_2'=z_1+z_2$；否则 $z_1'+z_2'=0$。具体过程如协议 1 所示。

协议 1　SRU 协议

输入　$\mathcal{S}_1$ 拥有 $z_1 \in \mathbb{Z}_n$，$\mathcal{S}_2$ 拥有 $z_2 \in \mathbb{Z}_n$

输出　$\mathcal{S}_1$ 返回 z_1'，$\mathcal{S}_2$ 返回 z_2'

1) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同执行 $b \leftarrow \mathrm{SComp}(z_1,z_2,0,0)$，计算 $\hat{b} \leftarrow 1-b$

2) $\mathcal{S}_1$ 计算并返回 $z_1' \leftarrow z_1\hat{b}$，$\mathcal{S}_2$ 计算并返回 $z_2' \leftarrow z_2\hat{b}$

在 SecRPN 中，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 执行安全逐点卷积获得目标边界框的判定分数，边界框内包含目标的概率为 s_{fg}，不包含目标的概率为 s_{bg}。Softmax 函数用于平衡边界框分数的分布，计算实质为 $\dfrac{\mathrm{e}^{s_j-\max\{s_j\}}}{\sum\limits_{j\in\{fg,bg\}}\mathrm{e}^{s_j-\max\{s_j\}}}$，若 $|j|=2$，则表现为 $\dfrac{\mathrm{e}^{s_j}}{\mathrm{e}^{s_{fg}}+\mathrm{e}^{s_{bg}}}$。针对该表达式提出一种 SST 协议，已知 $\{s_1',s_1''\}$ 和 $\{s_2',s_2''\}$，满足 $s_{fg}=s_1'+s_2'$ 和 $s_{bg}=s_1''+s_2''$，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同调用 SExp 协议计算 $\mathrm{e}^{s_1'+s_2'}$ 和 $\mathrm{e}^{s_1''+s_2''}$。由于表达式的分子和分母均是加法形式，考虑在不泄露隐私的前提下传递分母，通过传递加法结果 t_1 和 t_2，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 可以获得分母的值 t。最后，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 计算获得 $\{p_1',p_1''\}$ 和 $\{p_2',p_2''\}$，满足 $p_1'+p_2'=\dfrac{\mathrm{e}^{s_1'+s_2'}}{\mathrm{e}^{s_{fg}}+\mathrm{e}^{s_{bg}}}$ 和 $p_1''+p_2''=\dfrac{\mathrm{e}^{s_1''+s_2''}}{\mathrm{e}^{s_{fg}}+\mathrm{e}^{s_{bg}}}$，具体如协议 2 所示。

协议 2　SST 协议

输入　$\mathcal{S}_1$ 拥有 $\{s_1',s_1''\} \in \mathbb{Z}_n$，$\mathcal{S}_2$ 拥有 $\{s_2',s_2''\} \in \mathbb{Z}_n$

输出　$\mathcal{S}_1$ 返回 $\{p_1',p_1''\}$，$\mathcal{S}_2$ 返回 $\{p_2',p_2''\}$

1) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同执行 $t_1',t_2' \leftarrow \mathrm{SExp}(s_1',s_2')$ 和 $t_1'',t_2'' \leftarrow \mathrm{SExp}(s_1'',s_2'')$

2) $\mathcal{S}_1$ 计算 $t_1 \leftarrow t_1'+t_1''$ 并将 t_1 发送给 $\mathcal{S}_2$，$\mathcal{S}_2$ 计算 $t_2 \leftarrow t_2'+t_2''$ 并将 t_2 发送给 $\mathcal{S}_1$

3) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 计算 $t \leftarrow t_1+t_2$

4) $\mathcal{S}_1$ 计算并返回 $p_1' \leftarrow \dfrac{t_1'}{t}$ 和 $p_1'' \leftarrow \dfrac{t_1''}{t}$，$\mathcal{S}_2$ 计算并返回 $p_2' \leftarrow \dfrac{t}{t_2'}$ 和 $p_2'' \leftarrow \dfrac{t_2''}{t}$

4.3　安全锚变换模块

为了匹配不同的目标尺寸，SecRPN 需要构造一些锚作为基础边界框。若采用隐私保护 VGG-16（visual geometry group）网络[28]生成特征图，执行了 4 次步长为 2 的不重叠池化操作，这意味着此时特征图的尺寸只有原始图像的 $\dfrac{1}{16}$，因此生成的锚需要设置锚的最小间隔为 16，即基点坐标 $(0,0,15,15)$。在此基础上，设置锚的长宽比分别为

1:1、1:2 和 2:1，尺寸大小分别为 8、16 和 32，由这 3 种比例和 3 种尺寸组合而成的 9 种锚可以覆盖整个图像区域。显然，这些锚是粗糙的，不能直接用于选择目标位置。$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 可以利用逐点卷积生成的 2 路位移特征对这些锚执行变换操作，原始计算见 2.1 节。SAT 协议被设计用来实现该安全操作，已知 $\mathcal{S}_1$ 拥有锚坐标副本 $A_1=\{x_1',y_1',x_1'',y_1''\}$ 和位移值副本 $D_1=\{\mathrm{d}w_1,\mathrm{d}h_1,\mathrm{d}x_1,\mathrm{d}y_1\}$，$\mathcal{S}_2$ 拥有 $A_2=\{x_2',y_2',x_2'',y_2''\}$ 和 $D_2=\{\mathrm{d}w_2,\mathrm{d}h_2,\mathrm{d}x_2,\mathrm{d}y_2\}$，满足 $x_{\mathrm{up}}=x_1'+x_2'$、$y_{\mathrm{up}}=y_1'+y_2'$、$x_{\mathrm{bottom}}=x_1''+x_2''$ 和 $y_{\mathrm{bottom}}=y_1''+y_2''$，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 首先独立计算锚的宽 w_a、高 h_a 及中心坐标 (x_a,y_a)，满足 $w_a=A_1^w+A_2^w=x_{\mathrm{bottom}}-x_{\mathrm{up}}$、$h_a=A_1^h+A_2^h=y_{\mathrm{bottom}}-y_{\mathrm{up}}$、$x_a=A_1^x+A_2^x=x_{\mathrm{bottom}}+0.5\omega_a$ 和 $y_a=A_1^y+A_2^y=y_{\mathrm{bottom}}+0.5h_a$，然后协同执行 SMul 协议和 SExp 协议替换锚变换操作中的二元乘法和指数函数。SAT 协议的具体构造过程如协议 3 所示，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 经过步骤 2)～步骤 4)的交互计算得到目标检测框的宽、高及中心坐标，分别满足 $w_b=B_1^w+B_2^w$、$h_b=B_1^h+B_2^h$、$x_b=B_1^x+B_2^x$ 和 $y_b=B_1^y+B_2^y$，最后执行步骤 1)的逆操作获得目标边界框 B_i 的坐标 $\{B_i^{x'},B_i^{y'}B_i^{x''},B_i^{y''},i\in1,2\}$。

协议 3　SAT 协议

输入　$\mathcal{S}_1$ 拥有 $\{A_1,D_1\}\in\mathbb{Z}_n$，$\mathcal{S}_2$ 拥有 $\{A_2,D_2\}\in\mathbb{Z}_n$

输出　$\mathcal{S}_1$ 返回 B_1，$\mathcal{S}_2$ 返回 B_2

1) $\mathcal{S}_i\{i\in1,2\}$ 计算 $A_i^w\leftarrow x_i''-x_i'$，$A_i^h\leftarrow y_i''-y_i'$，$A_i^x\leftarrow x_i'+0.5A_i^w$，$A_i^y\leftarrow y_i'+0.5A_i^h$

2) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同执行 $e_1',e_2'\leftarrow\mathrm{SExp}(\mathrm{d}w_1,\mathrm{d}w_2)$ 和 $B_1^w,B_2^w\leftarrow\mathrm{SMul}(A_1^w,A_2^w,e_1',e_2')$

3) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同执行 $m_1',m_2'\leftarrow\mathrm{SMul}(A_1^w,A_2^w,\mathrm{d}x_1,\mathrm{d}x_2)$，$\mathcal{S}_i$ 计算 $B_i^x\leftarrow m_i'+A_i^x$

4) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 采用步骤 2 计算 B_i^h，采用步骤 3) 计算 B_i^y

5) $\mathcal{S}_i$ 计算 $B_i^{x'}\leftarrow B_i^x-0.5B_i^w$，$B_i^{x''}\leftarrow B_i^x+0.5B_i^w$，$B_i^{y'}\leftarrow B_i^y-0.5B_i^h$ 和 $B_i^{y''}\leftarrow B_i^y+0.5B_i^h$

6) $\mathcal{S}_1$ 返回 $B_1\leftarrow\{B_1^{x'},B_1^{y'},B_1^{x''},B_1^{y''}\}$；$\mathcal{S}_2$ 返回 $B_2\leftarrow\{B_2^{x'},B_2^{y'},B_2^{x''},B_2^{y''}\}$

4.4　安全边界框修正模块

经过安全锚变换操作，部分边界框超出了图像边界，或者因为尺寸太小不具有目标搜索价值，因此需要对这些边界框进行修正。在 SecRPN 中，裁剪操作负责将一些边界框限制在图像边界内，只保留图像区域内的部分，过滤操作则负责剔除小于单位大小的边界框。为了避免隐私信息泄露，本文设计了 SBC 协议和 SBF 协议。SBC 协议具体过程如协议 4 所示。已知 $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 分别拥有边界框副本 C_1 和 C_2，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同执行 SComp 协议判断 $0\leqslant C_1^{x'}+C_2^{x'}<W$ 是否成立，若 $C_1^{x'}+C_2^{x'}<0(b'=1)$，则限制 $C_1^{x'}+C_2^{x'}=0$；若 $C_1^{x'}+C_2^{x'}\geqslant W(b''=0)$，则限制 $C_1^{x'}+C_2^{x'}=W$。同理，其他坐标均按照 SBC 协议步骤进行裁剪，使边界框的横坐标 $0\leqslant C_1^{x'}+C_2^{x'}$，$C_1^{x''}+C_2^{x''}\leqslant W$，纵坐标 $0\leqslant C_1^{y'}+C_2^{y'},C_1^{y''}+C_2^{y''}\leqslant H$，进而，安全裁剪后的边界框 (E_1+E_2) 仅保留了 $W\times H$ 矩形区域内的部分。

协议 4　SBC 协议

输入　$\mathcal{S}_1$ 拥有 $C_1\in\mathbb{Z}_n$，$\mathcal{S}_2$ 拥有 $C_2\in\mathbb{Z}_n$，公共的图像边界宽 W 和高 H

输出　$\mathcal{S}_1$ 返回 E_1，$\mathcal{S}_2$ 返回 E_2

1) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 将 C_1 和 C_2 分别表示为 $\{C_1^{x'},C_1^{y'},C_1^{x''},C_1^{y''}\}$ 和 $\{C_2^{x'},C_2^{y'},C_2^{x''},C_2^{y''}\}$

2) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同执行 $b'\leftarrow\mathrm{SComp}(C_1^{x'},C_2^{x'},0,0)$ 和 $b''\leftarrow\mathrm{SComp}(C_1^{x'},C_2^{x'},W,0)$

3) if $b'=1$

4)　　$\mathcal{S}_1$ 计算 $C_1^{x'}\leftarrow0$，$\mathcal{S}_2$ 计算 $C_2^{x'}\leftarrow0$

5) end if

6) if $b''=0$

7)　　$\mathcal{S}_1$ 计算 $C_1^{x'}\leftarrow W$，$\mathcal{S}_2$ 计算 $C_2^{x'}\leftarrow0$

8) end if

9) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同执行步骤 2)～步骤 6)处理 $\{C_i^{y'},C_i^{x''},C_1^{y''}\}$

10) $\mathcal{S}_1$ 返回 $E_1\leftarrow\{C_1^{x'},C_1^{y'},C_1^{x''},C_1^{y''}\}$，$\mathcal{S}_2$ 返回 $E_2\leftarrow\{C_2^{x'},C_2^{y'},C_2^{x''},C_2^{y''}\}$

经过 SBC 协议计算，目标边界框的单一坐标数值可以控制在区间 $[0,W]$ 或 $[0,H]$，但不能判断边界框是否具有意义。若 $(C_1^{x''}+C_2^{x''})-(C_1^{x'}+C_2^{x'})<0$ 或 $(C_1^{y''}+C_2^{y''})-(C_1^{y'}+C_2^{y'})<0$，这意味着边界框的宽或高为负数，这种边界框显然不存在。同时，为了剔除过小的边界框，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同执行如协议 5 所示的 SBF 协议。已知边界框副本 F_1 和 F_2，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 独立计算横纵坐标的差值，利用 SComp 协议比较边

界框的宽 $w_1'+w_2'$ 与公共的边界框过滤阈值 η、高 $h_1'+h_2'$ 与 η 的大小关系，提取出符合条件 $w_1'+w_2'\geqslant\eta(a'=0)$ 和 $h_1'+h_2'\geqslant\eta(a''=0)$ 的边界框索引 a，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 进而可以获得过滤后的边界框副本 G_1 和 G_2。注意，SecRPN 中的阈值 η 为 16，该值与池化层次数有关。

协议 5 SBF 协议

输入 $\mathcal{S}_1$ 拥有 $F_1\in\mathbb{Z}_n$，$\mathcal{S}_2$ 拥有 $F_2\in\mathbb{Z}_n$，公共的边界框过滤阈值 η

输出 $\mathcal{S}_1$ 返回 G_1，$\mathcal{S}_2$ 返回 G_2

1) $\mathcal{S}_i(i=1,2)$ 初始化 $F_i\leftarrow\{F_i^{x'},F_i^{y'},F_i^{x''},F_i^{y''}\}$

2) $\mathcal{S}_i$ 计算 $w_i'\leftarrow F_i^{x''}-F_i^{x'}$ 和 $h_i'\leftarrow F_i^{y''}-F_i^{y'}$

3) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同计算 $a'\leftarrow\mathrm{SComp}(w_1',w_2',\eta,0)$ 和 $a''\leftarrow\mathrm{SComp}(h_1',h_2',\eta,0)$

4) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 计算 $a\leftarrow\mathrm{where}(a'=0\ \&\ a''=0)$

5) $\mathcal{S}_1$ 计算并返回 $G_1\leftarrow F_1(a)$，$\mathcal{S}_2$ 计算并返回 $G_2\leftarrow F_2(a)$

4.5 安全非极大值抑制模块

为了进一步缩减目标边界框的数量，提高位置检测效率，SecRPN 采用 NMS 方法来剔除相似的目标边界框，保留概率 $p_1'+p_2'$ 较高的目标边界框。IoU 用来描述 2 个边界框的相似程度，其定义为重叠区域面积与覆盖区域面积的比值，取值范围为[0,1]。如果 IoU=0，表示边界框无重叠；如果 IoU=1，表示边界框完全重叠。已知边界框 H 和 R 的位置关系如图 3 所示，则 IoU 可表示为 $\mathrm{IoU}_{H,R}=\dfrac{\Omega_{H\cap R}}{\Omega_H+\Omega_R-\Omega_{H\cap R}}$，其中，$\Omega_H$ 表示边界框 H 的面积，Ω_R 表示边界框 R 的面积，$\Omega_{H\cap R}$ 表示边界框 H 和边界框 R 的重叠区域面积。选择公共的相似阈值 η'，若 $\eta'\leqslant\mathrm{IoU}_{H,R}\leqslant1$，则认为边界框 H 和 R 是相似的。

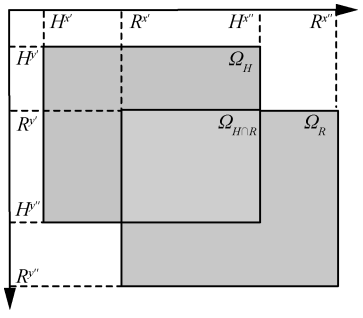


图 3 边界框 H 和 R 的位置关系

NMS 根据概率大小筛选相似的边界框，因此，在相似性判断之前，需要设计安全降序（SDS, secure descending sorting）协议，如协议 6 所示。已知边界框概率副本 P_1 和 P_2，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 相互传递 $\hat{P}_1$ 和 $\hat{P}_2$，获得的 $\hat{P}$ 与 P_1+P_2 具有相同的排列顺序，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 调用快速排序等方法可以对 $\hat{P}$ 进行降序排列，从而获得 P_1+P_2 的降序索引列表。

协议 6 SDS 协议

输入 $\mathcal{S}_1$ 拥有 $P_1\in\mathbb{Z}_n$，$\mathcal{S}_2$ 拥有 $P_2\in\mathbb{Z}_n$

输出 $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 返回 Υ

1) $\mathcal{S}_1$ 随机选择 $\rho_1\in\mathbb{Z}_n$，$\mathcal{S}_2$ 随机选择 $\rho_2\in\mathbb{Z}_n$

2) $\mathcal{S}_1$ 计算 $\hat{P}_1\leftarrow P_1-\rho_1$ 并将 $\hat{P}_1$ 发送给 $\mathcal{S}_2$，$\mathcal{S}_2$ 计算 $\hat{P}_2\leftarrow P_2-\rho_2$ 并将 $\hat{P}_2$ 发送给 $\mathcal{S}_1$

3) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 计算 $\hat{P}\leftarrow\hat{P}_1+\hat{P}_2$

4) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 执行降序运算 $\mathrm{sort}(\hat{P})$

5) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 将降序后的索引列表赋值给索引列表 Υ

在此基础上，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 协同执行 SNMS 协议，如协议 7 所示。已知目标边界框副本 U_1 和 U_2，以及其相应的概率副本 P_1 和 P_2，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 通过传递坐标差值间接地计算边界框的面积 S（步骤 2)～步骤 4)），而不会泄露完整的边界框坐标值。然后，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 调用 SDS 协议计算概率值降序排列后的索引列表 Υ（步骤 5)），保留概率最高的边界框索引 ∂（步骤 8)），将剩余边界框与之进行相似性判比，相似性问题可以归纳于重叠区域面积的安全计算。对于索引为 ∂ 和 k 的 2 个边界框，首先借助图 3 来判断两者的位置关系，比较左上角和右下角坐标值，获得重叠区域 T_1+T_2（步骤 11)～步骤 14)），接下来，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 计算 T_1+T_2 的面积 s，若 $s\leqslant0$，则认为 2 个边界框无重叠，此时 IoU = 0（步骤 15)～步骤 16)）。若 $\mathrm{IoU}\geqslant\eta'$，则认为边界框存在冗余，并从列表 Υ 中删除这些冗余索引（步骤 18)～步骤 19)）。经过若干次迭代分类边界框索引，直到 $\Upsilon\leftarrow\varnothing$，终止迭代。根据索引列表 Ψ，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 可以获得 NMS 抑制的边界框副本 V_1 和 V_2。

协议 7 SNMS 协议

输入 $\mathcal{S}_1$ 拥有 $U_1\in\mathbb{Z}_n$ 和 $P_1\in\mathbb{Z}_n$，$\mathcal{S}_2$ 拥有 $U_2\in\mathbb{Z}_n$ 和 $P_2\in\mathbb{Z}_n$，公共的相似阈值 η'

输出 $\mathcal{S}_1$ 返回 V_1，$\mathcal{S}_2$ 返回 V_2

1) $\mathcal{S}_i(i=1,2)$ 初始化 $U_i\leftarrow\{U_i^{x'},U_i^{y'},U_i^{x''},U_i^{y''}\}$

2) $\mathcal{S}_1$ 计算 $U_1^w\leftarrow U_1^{x''}-U_1^{x'}$ 和 $U_1^h\leftarrow U_1^{y''}-U_1^{y'}$，

并将U_1^w和U_1^h发送给$\mathcal{S}_2$

3) $\mathcal{S}_2$计算$U_2^w \leftarrow U_2^{x''} - U_2^{x'}$和$U_2^h \leftarrow U_2^{y''} - U_2^{y'}$，并将$U_2^w$和$U_2^h$发送给$\mathcal{S}_1$

4) $\mathcal{S}_1$和$\mathcal{S}_2$计算$S \leftarrow (U_1^w + U_2^w)(U_1^h + U_2^h)$

5) $\mathcal{S}_1$和$\mathcal{S}_2$协同计算$\Upsilon \leftarrow \mathrm{SDS}(P_1, P_2)$

6) $\mathcal{S}_1$和$\mathcal{S}_2$创建索引列表Ψ

7) where $\Upsilon \neq \varnothing$

8) $\mathcal{S}_1$和$\mathcal{S}_2$初始化$\partial \leftarrow \Upsilon[0]$，将$\partial$添加至$\Psi$末尾，并将$\partial$从$\Upsilon$中移除

9) for k in Υ

10) $\mathcal{S}_1$赋值$T_1 \leftarrow U_1[\partial]$；$\mathcal{S}_2$赋值$T_2 \leftarrow U_2[\partial]$

11) $\mathcal{S}_1$和$\mathcal{S}_2$协同计算$\rho' \leftarrow \mathrm{SComp}(U_1^{x'}[\partial], U_2^{x'}[\partial], U_1^{x'}[k], U_2^{x'}[k])$

12) if $\rho' = 1$

13) $\mathcal{S}_1$赋值$T_1^{x'} \leftarrow U_1^{x'}[k]$；$\mathcal{S}_2$赋值$T_2^{x'} \leftarrow U_2^{x'}[k]$

14) $\mathcal{S}_1$和$\mathcal{S}_2$协同按照步骤 11)～步骤 13)方法计算$\{T_i^{y'}, T_i^{x''}, T_i^{y''}, i = 1,2\}$

15) $\mathcal{S}_1$和$\mathcal{S}_2$按照步骤 2)～步骤 4)计算边界框$T_1 + T_2$的面积s

16) $\mathcal{S}_1$和$\mathcal{S}_2$计算$s' \leftarrow \max(s, 0)$和$\mathrm{IoU} \leftarrow \dfrac{s'}{S[\partial] + S[k] - s'}$

17) end if

18) if $\mathrm{IoU} \geqslant \eta'$

19) $\mathcal{S}_1$和$\mathcal{S}_2$并将k从Υ中移除

20) end if

21) end for

22) end where

23) $\mathcal{S}_1$计算并返回$V_1 \leftarrow U_1(\Psi)$，$\mathcal{S}_2$计算并返回$V_2 \leftarrow U_2(\Psi)$

5　理论分析

5.1　正确性分析

已知特征图副本x_1和x_2，$\mathcal{S}_1$和$\mathcal{S}_2$执行 SecRPN 后输出目标边界框V_1和V_2，其正确性依赖于所设计的安全协议。在之前的工作[24, 28]中，SMul 协议、SComp 协议、SExp 协议已经被证明是正确的，具体构造过程见附录。在特征处理模块中，$\omega(x_1 + x_2) + b$通过卷积计算可以线性地拆分为$\omega x_1 + b$和$\omega x_2 + b$，由 SRU 协议完成 ReLU 激活计算，$\mathcal{S}_1$和$\mathcal{S}_2$利用 SComp 协议比较$z_1 + z_2$和 0 的大小，进而选择将输入置 0 或维持不变。在 SAT 协议中，$\mathcal{S}_1$和$\mathcal{S}_2$利用 SMul 协议和 SExp 协议来正确计算出$B_1^w + B_2^w = (A_1^w + A_2^w)e^{\mathrm{d}w_1 + \mathrm{d}w_2}$、$B_1^h + B_2^h = (A_1^h + A_2^h)e^{\mathrm{d}h_1 + \mathrm{d}h_2}$、$B_1^x + B_2^x = (A_1^x + A_2^x) + (A_1^w + A_2^w) \cdot (\mathrm{d}x_1 + \mathrm{d}x_2)$和$B_1^y + B_2^y = (A_1^y + A_2^y) + (A_1^h + A_2^h)(\mathrm{d}y_1 + \mathrm{d}y_2)$，这与原始的锚变换计算是一致的。在 SBC 协议中，$\mathcal{S}_1$和$\mathcal{S}_2$利用 SComp 协议来判断横坐标与区间$[0, W]$、纵坐标与区间$[0, H]$的长度关系，获得边界框$C_1 + C_2$与矩形区域$\{0,0\} \rightarrow \{W, H\}$的位置关系。若存在重叠，则$\mathcal{S}_1$和$\mathcal{S}_2$输出的边界框$E_1 + E_2$为重叠部分；否则$E_1 + E_2$仅为点$(0,0)$。在 SBF 协议中，$\mathcal{S}_1$和$\mathcal{S}_2$计算得到边界框的宽$w_1' + w_2'$和高$h_1' + h_2'$，利用 SComp 协议可以正确判断出两者与阈值$\eta$的长度关系，进而获得宽和高均不小于$\eta$的边界框$G_1 + G_2$。在 SNMS 协议中，$\mathcal{S}_1$和$\mathcal{S}_2$利用 SDS 协议对$P_1 + P_2$执行降序操作，由于$\hat{P}$的每个元素与$P_1 + P_2$均相差常数$\rho_1 + \rho_2$，$\mathcal{S}_1$和$\mathcal{S}_2$可以直接对$\hat{P}$执行降序操作。$U_1[\partial] + U_2[\partial]$是概率最高的边界框，$\mathcal{S}_1$和$\mathcal{S}_2$计算其与剩余边界框的相似度 IoU，其中边界框重叠区域$T_1 + T_2$的计算类似于 SBC 协议，可以利用 SComp 协议比较并获得坐标以及区域面积，$\mathcal{S}_1$和$\mathcal{S}_2$保留 IoU 低于阈值η'的边界框，经过多轮迭代，$\mathcal{S}_1$和$\mathcal{S}_2$可以正确输出抑制处理后的边界框V_1和V_2。显然，这一系列计算协议可以保证 SecRPN 在理论上是完全正确的，但在实际计算中，数值精度等因素会引入误差，经过验证，SecRPN 的计算结果误差可以维持在10^{-5}左右，性能评估见 6.1 节。

5.2　安全性分析

在半可信模型中，假设存在概率多项式时间的模拟器$\mathcal{M}$，为敌手$\mathcal{A}$生成一组模拟视图，若该视图在计算上与真实视图无法区分，则认为提出的计算协议是安全的。在安全性证明之前，需要引入下述引理[28-30]。

引理 1　若协议调用的所有子协议在概率多项式时间内是可模拟的，那么该协议是可模拟的。

引理 2　若$a \in \mathbb{Z}_m$是均匀分布的，并且与$b \in \mathbb{Z}_m$相互独立，那么认为$a \pm b$也是均匀分布的，并且与b相互独立。

根据引理 1 所述，SecRPN 的安全性可以归结于所设计协议的安全性证明。其中，SMul、SComp 协议和 SExp 协议的子协议的安全性在文献[22, 26]

中已经得到证明。同时，所设计协议以数组作为输入，由于每个数组元素的计算形式一致，根据引理 2 可知，参与计算的随机数组也是均匀分布的。

定理 1 在半可信模型中，SRU 协议、SST 协议和 SAT 协议是安全的。

证明 在 SRU 协议中，$\mathcal{S}_1$ 的真实视图为 $\{z_1, b, \hat{b}, z_1'\}$，由 SComp 协议得到的 b 和 $\hat{b}$ 仅表示比较符号信息，模拟器 $\mathcal{M}$ 为 $\mathcal{S}_1$ 生成均匀随机分布的模拟视图，敌手 $\mathcal{A}$ 在计算意义上无法区分两者。同理，敌手 $\mathcal{A}$ 也无法区分 $\mathcal{S}_2$ 的真实视图 $\{z_2, b, \hat{b}, z_2'\}$ 及其模拟试图。在 SST 协议中，$\mathcal{S}_1$ 的真实视图为 $\{s_1', s_1'', t_1', t_1'', t_1, t_2, t, p_1', p_1''\}$，由 SExp 协议得到的 t_1' 和 t'' 不能直接发送给 $\mathcal{S}_2$，但两者的差值 t_1 是均匀分布的，将其作为传递内容可以获得 Softmax 函数公共分母的值 t，模拟器 $\mathcal{M}$ 为 $\mathcal{S}_1$ 生成均匀的模拟视图，敌手 $\mathcal{A}$ 在计算意义上无法区分。同理，敌手 $\mathcal{A}$ 也无法区分 $\mathcal{S}_2$ 的真实视图和模拟视图。在 SAT 协议中，$\mathcal{S}_i(i=1,2)$ 的真实视图为 $\{A_i, D_i, e_i', m_i', B_i\}$，中间数值由交互执行 SExp 协议和 SMul 协议获得，根据引理 1，SAT 协议的安全性可以由 SExp 协议和 SMul 协议的子协议提供保证，敌手 $\mathcal{A}$ 在计算意义上也无法区分真实视图与模拟器 $\mathcal{M}$ 随机生成的模拟试图。因此，SRU 协议、SST 协议和 SAT 协议是安全的。

证毕。

定理 2 在半可信模型中，SBC 协议、SBF 协议、SDS 协议和 SNMS 协议是安全的。

证明 在 SBC 协议中，$\mathcal{S}_i(i=1,2)$ 的真实视图为 $\{C_i, b', b'', c', c'', E_i\}$，由 SComp 协议获得的符号值 $\{b', b'', c', c''\}$ 是均匀分布的随机值，在计算中公开不会泄露隐私副本，模拟器 $\mathcal{M}$ 负责为 $\mathcal{S}_i$ 生成模拟视图，敌手 $\mathcal{A}$ 在计算上无法将其与真实视图进行区分。在 SBF 协议中，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 利用 SComp 协议判断 $w_1' + w_2' \geqslant \eta$ 和 $h_1' + h_2' \geqslant \eta$ 是否成立，并通过公开符号值 $\{a', a''\}$ 实现索引 a 的取舍，敌手 $\mathcal{A}$ 仍然无法区分模拟器 $\mathcal{M}$ 为 $\mathcal{S}_i$ 生成的模拟视图与其真实视图 $\{F_i, b', a', a'', a, G_i\}$。在 SDS 协议中，$\mathcal{S}_1$ 的真实视图为 $\{P_1, \rho_1, \hat{P}_1, \hat{P}_2, \hat{P}, \Upsilon\}$，其中 ρ_1 是从 $\mathbb{Z}_n$ 内随机选择的。根据引理 2 可知，$\hat{P}_1 \leftarrow P_1 - \rho_1$ 是均匀分布的，$\mathcal{S}_2$ 不能推测出 $\mathcal{S}_1$ 的隐私信息 P_1。若模拟器 $\mathcal{M}$ 为 $\mathcal{S}_1$ 生成相应的模拟视图，敌手 $\mathcal{A}$ 在计算上无法进行区分模拟和真实视图。同理，$\hat{P}_2$ 也是均匀分布的，敌手 $\mathcal{A}$ 在计算上仍然无法区分 $\mathcal{S}_2$ 的模拟和真实视图。在 SNMS 协议中，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 通过传递边界框宽和高计算面积，然而，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 仅仅获知长度特征并不能确定边界框的位置，边界框的完整坐标值仍然是安全的。SDS 协议负责边界框概率的排序，2 个边界框的重叠区域及其 IoU 计算类似于 SBC 协议，由于 SComp 协议、SBC 协议、SDS 协议已经被证明是安全的，存在模拟器 $\mathcal{M}$ 为 $\mathcal{S}_i$ 生成均匀的模拟视图，敌手 $\mathcal{A}$ 在计算上无法将其与真实视图 $\{U_i, P_i, S, \Upsilon, \Psi, T_1, s, s', \text{IoU}\}$ 进行区分。因此，SBC 协议、SBF 协议、SDS 协议和 SNMS 协议是安全的。

证毕。

5.3 复杂度分析

本节从计算复杂度和通信复杂度两方面评估和分析所提 SecRPN 中安全计算协议的效率，其结果分别如表 1 和表 2 所示。从表 1 可以看出，相比于明文环境下的 RPN[17]，引入安全计算协议后显然会增加计算开销。在 SRU 协议中，SecRCNN[25]采用的比较协议[24]与数值的二进制位长度相关，而 SecRPN 基于顺序结构的 SComp 协议可以完成激活计算，其计算复杂度为 $\mathcal{O}(N)$。在 SST 协议中，相比于 SecRCNN[25]，SecRPN 调用的 SExp 协议不需要多轮迭代，并且采用传递公共分母方式可以避免额外的比较计算，其计算复杂度为 $\mathcal{O}(N)$。关于 SAT 协议、SBC 协议和 SBF 协议，由于 SecRPN 底层的 SComp 协议是顺序执行的，因此计算复杂度均与 RPN[17]相同。关于 SDS 协议，SecRCNN[25]的每一次比较均需要调用比较协议，而 SecRPN 的 SDS 协议与 RPN[17]的快速排序相似，其计算复杂度为 $\mathcal{O}(N\log N)$。此外，SecRPN 的 SNMS 协议顺序执行 SDS 协议和计算 IoU 的循环结构，其计算复杂度为 $\mathcal{O}(N\log N)$。

表 1 安全计算协议的计算复杂度

协议	RPN[17]	SecRCNN[25]	SecRPN
SRU	$\mathcal{O}(N)$	$\mathcal{O}(NL)$	$\mathcal{O}(N)$
SST	$\mathcal{O}(N)$	$\mathcal{O}(NML)$	$\mathcal{O}(N)$
SAT	$\mathcal{O}(N)$	—	$\mathcal{O}(N)$
SBC	$\mathcal{O}(N)$	—	$\mathcal{O}(N)$
SBF	$\mathcal{O}(N)$	—	$\mathcal{O}(N)$
SDS	$\mathcal{O}(N\log N)$	$\mathcal{O}(NL\log N)$	$\mathcal{O}(N\log N)$
SNMS	$\mathcal{O}(N\log N)$	$\mathcal{O}(N^2L)$	$\mathcal{O}(N\log N)$

注：N 表示数组长度；L 表示二进制位长度；M 表示迭代次数。

表 2 描述了协议中 $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 之间的通信轮数及

其通信开销，底层的SMul协议、SExp协议和SComp协议均不依赖任何循环操作，仅需要一轮或3轮通信。在SecRPN中，SRU协议执行一次SComp协议，SST协议执行2次SExp协议和一次数据传递，均需要3轮通信。在SAT协议中，包含4次乘法和2次指数计算，共需要6轮通信。SBC和SBF分别执行8次和2次SComp协议，分别需要24轮和6轮通信。SDS协议仅需要一轮通信传递虚构概率副本，SNMS协议的通信开销依赖于IoU循环次数，需要$2+13\log N$轮通信。

表 2　　安全计算协议的通信复杂度

协议	通信轮数/轮	通信开销	协议	通信轮数/轮	通信开销
SMul	1	$2\|\|N\|\|$	SAT	6	$10\|\|N\|\|$
SExp	1	$\|\|N\|\|$	SBC	24	$32\|\|N\|\|$
SComp	3	$4\|\|N\|\|$	SBF	6	$8\|\|N\|\|$
SRU	3	$4\|\|N\|\|$	SDS	1	$\|\|N\|\|$
SST	3	$3\|\|N\|\|$	SNMS	$2+13\log N$	$3\|\|N\|\|+18\|\|N\|\|\log N$

注：$\|\|N\|\|$表示单位消息大小。

6　性能评估

本节将对所提安全计算协议和SecRPN的实际性能进行评估，同时分析验证基于SecRPN的目标检测结果的正确性和安全性。本文实验环境为Intel(R) Core(TM) i7-8565U CPU@1.80 GHz，20 GB RAM硬件配置的64位计算机，在Pycharm仿真平台上进行实验，利用Numpy工具完成协议的数组传递和计算。

6.1　安全计算协议性能

本文将从实际计算开销、通信开销和计算误差3个方面评估安全计算协议的性能。通过断点测试，利用安全计算协议的运行时间衡量计算开销，服务器之间传输的数据量大小衡量负载的通信开销，将安全计算协议与明文函数的输出结果的最大差异作为计算误差。批处理大小（数组长度N）是影响计算和通信开销的主要因素，从图4(a)～图4(f)可知，安全计算协议的计算开销随着批处理的增大而增加。相比于文献[22]中的安全ReLU协议，当$N=10^5$时，SRU协议的计算效率提高了近30倍，其时间开销约为ReLU计算的4倍（图4(a)）。SST和SAT协议的时间开销与原始的Softmax函数和锚变换操作相比较，时间开销没有明显增加，若利用泰勒展开方式[32]设计这2种安全协议，时间开销会随着迭代次数m而增加，且远高于本文提出的协议（图4(b)和图4(c)）。相比于明文环境下RPN的边界框裁剪、边界框过滤和NMS计算，SBC协议、SBF协议和SNMS协议处理长度为10^4以内的数组，计算开销没有明显增长趋势，即使处理长度为10^5的数组，计算开销也可以分别控制在95 ms、410 ms和1 150 ms内（图4(d)～图4(f)）。

由图4(g)和图4(h)可知，计算协议的通信开销随着数组长度N的增大而增加，当处理长度为10^5的数组，SRU协议、SST协议、SAT协议和SBF协议的通信开销控制在4 MB内，SBC协议和SNMS协议的通信轮数相对频繁，其通信开销也可以控制在15 MB内。输入范围是影响计算误差的主要因素。图4(i)显示当输入在1～20时，SST协议的计算误差维持在10^{-5}量级，等同于迭代40次的SecST协议。从图4(j)可知，SAT协议的计算误差与锚的坐标值范围和位移值范围有关，当位移大于10后，SAT协议的计算误差增长比较缓慢，且远小于SecAT协议。由于SComp协议的计算误差不会影响到整数部分，因此输出的符号位不会影响比较结果，不考虑系统抖动因素，在此基础上设计的SRU协议、SBC协议、SBF协议和SNMS协议可以实现零误差。

6.2　目标检测结果

本文采用数据集PASCAL VOC 2007 [31]进行实验，该数据集包含20个类别，共9 963张图片（其中，5 011张为训练图片，4 952张为测试图片），超过27 000个目标边界框。随机挑选一张图片，$\mathcal{S}_1$和$\mathcal{S}_2$利用设计的安全计算协议交互执行SecRPN，开销如表3所示，执行512条通道的安全卷积操作需要52.0 ms，相应的安全ReLU激活需要178.5 ms，然后利用大小为1×1的卷积核获得目标位移值和分数需要0.6 ms，针对目标和背景2个类别执行安全Softmax操作需要3.0 ms。根据9种不同比例和尺寸的锚，执行安全锚变换、安全边界框修正（裁剪和过滤）及安全NMS操作均可以维持在毫秒级。在SecRPN中，因数值精度引入的计算误差如图5(a)所示，SST协议的计算误差约为10^{-8}，后续协议操作的计算误差可以维持在10^{-5}量级。综上所述，SecRPN实际产生的计算开销为340.7 ms，约为明文环境下RPN的4倍，通信开销为27.21 MB，均优于现有工作[25]，

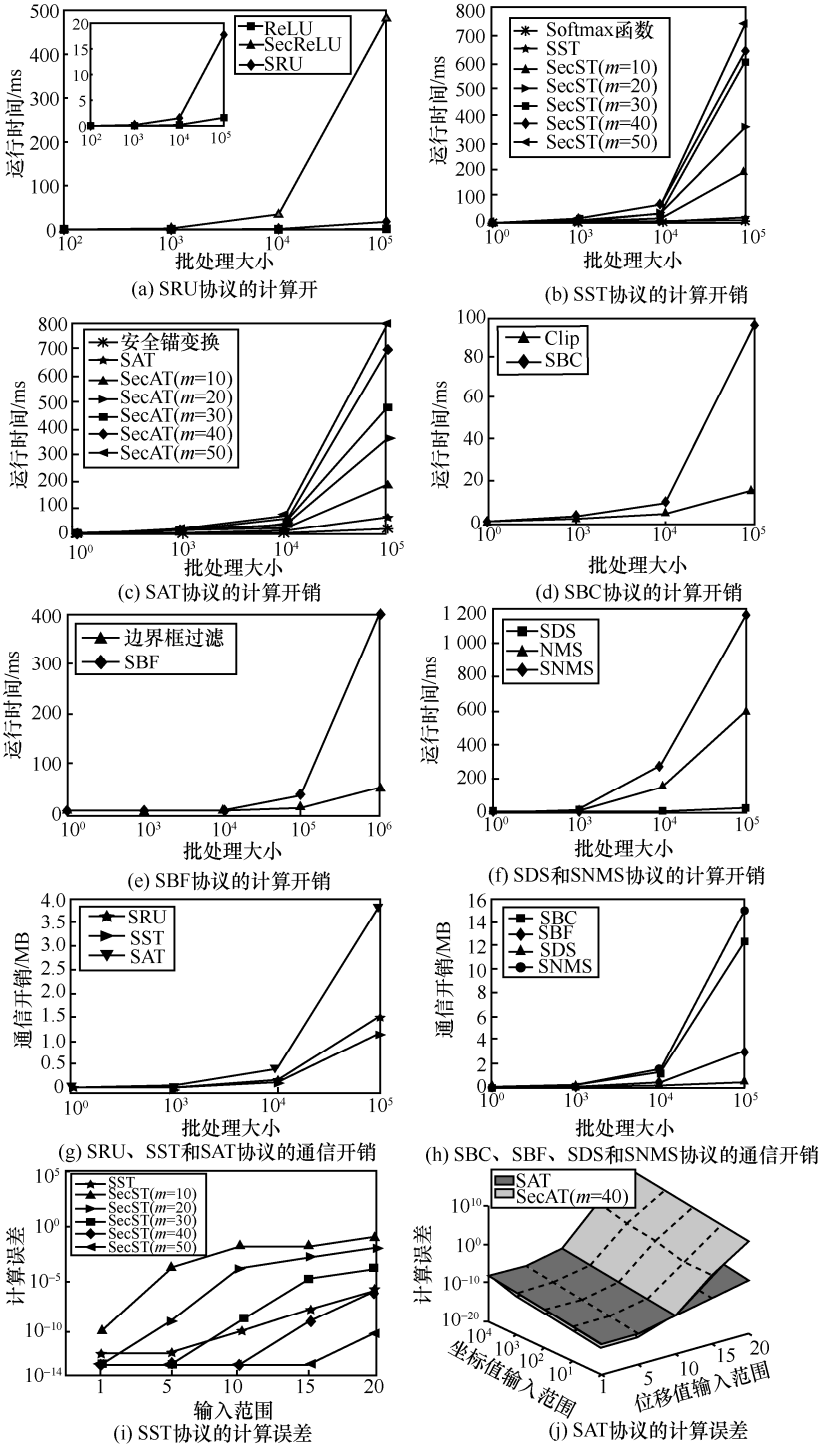


图 4 安全计算协议性能结果

具体如表 4 所示。

随机挑选的图片经过 SecRPN 处理后，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 将各自的目标边界框副本和分类概率向量发送给 $\mathcal{P}$，如图 5(b)所示，$\mathcal{P}$ 利用加法可以恢复出完整的目标检测边界框[80,46,422,298]和预测类别“bus”，并且与明文环境下的目标检测边界框仅存在10^{-5} 误差。图 5(c)显示，相比于正确的目标边界框，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 获得的边界框副本[374,501,2974,1 038]和[–294, –455, –2 552, –740]是无意义的。特别地，为了进一步凸显 SecRPN 的安全性，下面提出一种测试方法。$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 独自地进行检测操作，重复独立执行 3 次后，检测结果分别如图 5(d)和 5(e)所示，相比于正确

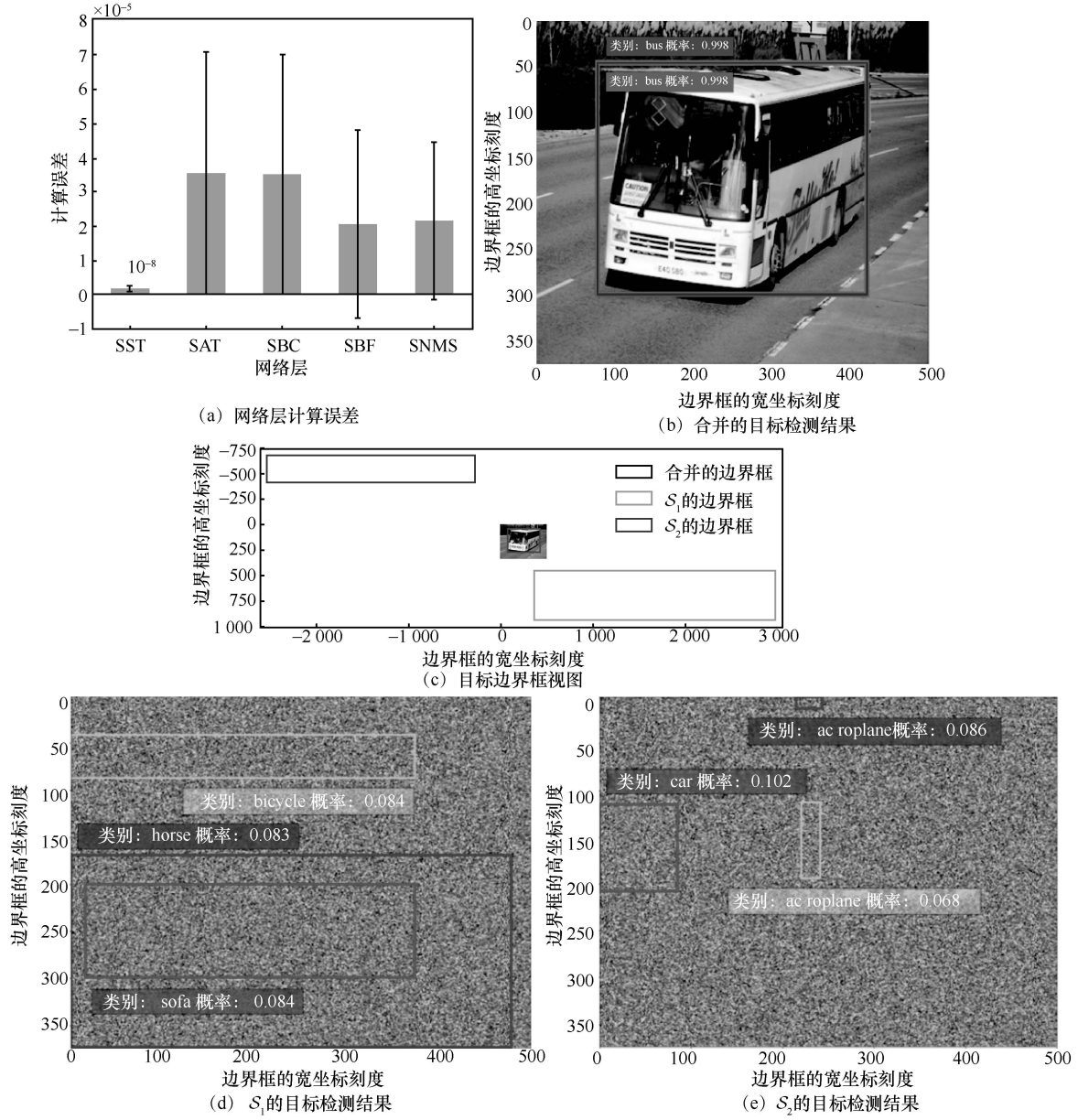
表 3　　网络层的开销比较

网络层	计算开销/ms		倍数/倍	通信开销/MB
	RPN[17]	SecRPN		
3×3 卷积	27.3	52.0	1.9	0
安全 ReLU	20.9	178.5	8.5	14.84
逐点卷积	0.3	0.6	2.0	0
Softmax	0.9	3.0	3.3	0.39
锚变换	2.0	9.0	4.5	0.65
边界框裁剪	3.2	15.6	4.9	8.35
边界框过滤	1.0	2.1	2.1	2.09
NMS	24.8	59.8	2.4	0.89

的目标边界框和类别，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 获得的目标类别和位置结果是随机的，并且目标概率近似于均匀分布，约为 0.05。由此可见，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 及约束下的敌手 $\mathcal{A}$ 均无法获得正确的检测结果，证明了 SecRPN 是正确且安全的。

表 4　　网络的开销比较

网络	计算开销/s	倍数/倍	通信开销/MB
RPN	0.086 2	—	—
SecRCNN	5.776 0	67.0	37.66
SecRPN	0.340 7	4.0	27.21



（a）网络层计算误差　（b）合并的目标检测结果

（c）目标边界框视图

（d）$\mathcal{S}_1$的目标检测结果　（e）$\mathcal{S}_2$的目标检测结果

图 5　SecRPN 的检测结果

7 结束语

针对物联网外包环境下目标检测任务的图像隐私泄露问题，本文在双边缘协作模式下基于加性秘密共享方案设计了一系列安全计算协议，组合的 SecRPN 可以在保证目标特征和位置隐私性的前提下实现目标检测。完备的理论分析证明了安全计算协议和 SecRPN 的正确性、安全性和高效性，实验结果表明 SecRPN 仅耗费 0.34 s 的时间成本，边缘节点之间需要负载 27.21 MB 的通信开销，并且计算误差可以控制在10^{-5}左右，这对于实时需求严苛的物联网服务具有良好的应用前景。在未来工作中，将继续研究降低隐私目标检测任务开销和误差的解决方法。

附录 协议的构造过程

在 2.2 节中，本文简单描述了 SMul、SExp 和 SComp 协议的功能，下面，将具体给出协议的完整构造过程。如协议 8 所示，SMul 协议利用可信第三方服务器产生的乘法三元组 $\{a,b,c\}$ 建立加法和乘法运算的联系，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 结合加性秘密共享可以协同计算乘法多项式 $(u_1+u_2)(v_1+v_2)\leftarrow(\alpha+a)(\beta+b)$。在协议 9 中，可信第三方服务器加法拆分乘法因式 $g_1+g_2\leftarrow d_1d_2$，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 可以基于这种变换思想执行 SExp 协议，协同计算 $\mathrm{e}^{u_1'}\mathrm{e}^{u_2'}\leftarrow(\vartheta_1+d_1)(\vartheta_2+d_2)$。SComp 协议如协议 10 所示，要比较 $u_1''+u_2''$ 与 $v_1''+v_2''$ 的大小，相当于判断 $(u_1''-v_1'')+(u_2''-v_2'')$ 的正负性，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 利用 $\mathcal{T}$ 生成的 $q_1+q_2=r_1r_2$ 联系，$\mathcal{S}_1$ 将加法副本隐藏至 θ_1 中传递给 $\mathcal{S}_2$，$\mathcal{S}_2$ 随机生成的非零分母使 $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 可以协同计算 $\phi_1\phi_2=(u_1''-v_1'')+(u_2''-v_2'')$，进一步地，$\mathcal{S}_1$ 和 $\mathcal{S}_2$ 将计算结果与正数 σ_1 相乘后相互传递，比较结果取决于 $\phi_1\varphi_2$ 和 $\phi_2\varphi_1$ 的正负性。

协议 8 SMul 协议

输入 $\mathcal{S}_1$ 拥有 $u_1,v_1\in\mathbb{Z}_n$，$\mathcal{S}_2$ 拥有 $u_2,v_2\in\mathbb{Z}_n$

输出 $\mathcal{S}_1$ 返回 f_1，$\mathcal{S}_2$ 返回 f_2

1) $\mathcal{T}$ 随机生成 $a,b\in\mathbb{Z}_n$，计算 $c\leftarrow ab$

2) $\mathcal{T}$ 随机拆分 $a\leftarrow a_1+a_2$，$b\leftarrow b_1+b_2$ 和 $c\leftarrow c_1+c_2$，并将 a_i，b_i 和 c_i 发送给 $\mathcal{S}_i(i=1,2)$

3) $\mathcal{S}_1$ 计算 $\alpha_1\leftarrow u_1-a_1$ 和 $\beta_1\leftarrow v_1-b_1$，并将 α_1 和 β_1 发送给 $\mathcal{S}_2$

4) $\mathcal{S}_2$ 计算 $\alpha_2\leftarrow u_2-a_2$ 和 $\beta_2\leftarrow v_2-b_2$，并将 α_2 和 β_2 发送给 $\mathcal{S}_1$

5) $\mathcal{S}_1$ 和 $\mathcal{S}_2$ 计算 $\alpha\leftarrow\alpha_1+\alpha_2$ 和 $\beta\leftarrow\beta_1+\beta_2$

6) $\mathcal{S}_1$ 计算并返回 $f_1\leftarrow c_1+b_1\alpha+a_1\beta$，$\mathcal{S}_2$ 计算返回 $f_2\leftarrow c_2+b_2\alpha+a_2\beta+\alpha\beta$

协议 9 SExp 协议

输入 $\mathcal{S}_1$ 拥有 $u_1'\in\mathbb{Z}_n$，$\mathcal{S}_2$ 拥有 $u_2'\in\mathbb{Z}_n$

输出 $\mathcal{S}_1$ 返回 f_1'，$\mathcal{S}_2$ 返回 f_2'

1) $\mathcal{T}$ 随机生成 $d_1,d_2\in\mathbb{Z}_n$，计算 $g\leftarrow d_1d_2$

2) $\mathcal{T}$ 随机拆分 $g\leftarrow g_1+g_2$，并将 d_i 和 g_i 发送给 $\mathcal{S}_i(i=1,2)$

3) $\mathcal{S}_1$ 计算 $\vartheta_1\leftarrow e^{u_1'}-d_1$ 并将 ϑ_1 发送给 $\mathcal{S}_2$，$\mathcal{S}_2$ 计算 $\vartheta_2\leftarrow e^{u_2'}-d_2$ 并将 ϑ_2 发送给 $\mathcal{S}_1$

4) $\mathcal{S}_1$ 计算并返回 $f_1'\leftarrow g_1+d_1\vartheta_2$，$\mathcal{S}_2$ 计算返回 $f_2'\leftarrow g_2+d_2\vartheta_1+\vartheta_1\vartheta_2$

协议 10 SComp 协议

输入 $\mathcal{S}_1$ 拥有 $u_1'',v_1''\in\mathbb{Z}_n$，$\mathcal{S}_2$ 拥有 $u_2'',v_2''\in\mathbb{Z}_n$

输出 $\mathcal{S}_1$ 返回 f''，$\mathcal{S}_2$ 返回 f''

1) $\mathcal{T}$ 随机生成 $r_1,r_2\in\mathbb{Z}_n$，计算 $q\leftarrow r_1r_2$

2) $\mathcal{T}$ 随机拆分 $q\leftarrow q_1+q_2$，并将 r_i 和 q_i 发送给 $\mathcal{S}_i(i=1,2)$

3) $\mathcal{S}_1$ 计算 $\theta_1\leftarrow u_1''-v_1''-r_1$，并将 θ_1 发送给 $\mathcal{S}_2$，$\mathcal{S}_2$ 计算 $\theta_2\leftarrow u_2''-v_2''$

4) $\mathcal{S}_2$ 随机选择 $\phi_2\in\mathbb{Z}_n^*$，计算 $\lambda\leftarrow\frac{\theta_1+\theta_2}{\phi_2}-q_2$ 和 $\mu\leftarrow\frac{1}{\phi_2}+r_2$，将 λ 和 μ 发送给 $\mathcal{S}_1$

5) $\mathcal{S}_1$ 计算 $\phi_1\leftarrow\lambda+r_1\mu-q_1$

6) $\mathcal{S}_1$ 随机选择 $\sigma_1\in\mathbb{Z}_n^*$，计算 $\varphi_1\leftarrow\sigma_1\phi_1$，并将 φ_1 发送给 $\mathcal{S}_2$

7) $\mathcal{S}_2$ 随机选择 $\sigma_2\in\mathbb{Z}_n^*$，计算 $\varphi_2\leftarrow\sigma_2\phi_2$，并将 φ_2 发送给 $\mathcal{S}_1$

8) $\mathcal{S}_1$ 计算并返回 $f''\leftarrow\mathrm{sign}(\phi_1\varphi_2)$，$\mathcal{S}_2$ 计算返回 $f''\leftarrow\mathrm{sign}(\phi_2\varphi_1)$

参考文献：

[1] LIU Y, MA X, SHU L, et al. Internet of things for noise mapping in smart cities: state-of-the-art and future directions[J]. IEEE Network, 2020, 34(4): 112-118.

[2] XIONG J, REN J, CHEN L, et al. Enhancing privacy and availability for data clustering in intelligent electrical service of IoT[J]. IEEE Internet of Things Journal, 2019, 6(2): 1530-1540.

[3] 艾瑞咨询. 中国智能物联网（AIoT）白皮书[R]. (2020-02-28)[2020-07-20].
Iresearch. White paper on China's artificial intelligent and Internet of things (AIoT)[R]. (2020-02-28)[2020-07-20].

[4] XIONG Z, LI W, HAN Q, et al. Privacy-preserving auto-driving: a GAN-Based approach to protect vehicular camera data[C]//2019 IEEE International Conference on Data Mining. Piscataway: IEEE Press, 2019: 668-677.

[5] PÉREZ-HERNÁNDEZ F, TABIK S, LAMAS A, et al. Object detection binary classifiers methodology based on deep learning to identify

small objects handled similarly: application in video surveillance[J]. Knowledge-Based Systems, 2020(194): 105590.

[6] XIONG J B, CHEN X, YANG Q, et al. A task-oriented user selection incentive mechanism in edge-aided mobile crowdsensing[J]. IEEE Transactions on Network Science and Engineering, 2019, doi: 10.1109/TNSE. 2019.2940958.

[7] CHEN Q, TANG S, YANG Q, et al. Cooper: cooperative perception for connected autonomous vehicles based on 3d point clouds[C]//2019 IEEE 39th International Conference on Distributed Computing Systems.Piscataway : IEEE Press, 2019: 514-524.

[8] CHEN Q, MA X, TANG S, et al. F-cooper: feature based cooperative perception for autonomous vehicle edge computing system using 3D point clouds[C]//4th ACM/IEEE Symposium on Edge Computing. New York: ACM Press, 2019: 88-100.

[9] LIN L, LIAO X, JIN H, et al. Computation offloading toward edge computing [J]. Proceedings of the IEEE, 2019, 107(8): 1584-1607.

[10] WANG X, HAN Y, LEUNG V, et al. Convergence of edge computing and deep learning: a comprehensive survey[J]. IEEE Communications Surveys & Tutorials, 2020, 22(2): 869-904.

[11] ZHAO Z, ZHENG P, XU S, et al. Object detection with deep learning: a review[J]. IEEE Transactions on Neural Networks and Learning Systems, 2019, 30(11): 3212-3232.

[12] REN J, GUO Y, ZHANG D, et al. Distributed and efficient object detection in edge computing: challenges and solutions[J]. IEEE Network, 2018, 32(6): 137-143.

[13] NIKOUEI S, CHEN Y, SONG S, et al. Real-time human detection as an edge service enabled by a lightweight CNN[C]//2018 IEEE International Conference on Edge Computing. Piscataway: IEEE Press, 2018: 125-129.

[14] ZHANG H, ZHANG Z, ZHANG L, et al. Object tracking for a smart city using IoT and edge computing[J]. Sensors, 2019, 19(9): 1987-2009.

[15] GIRSHICK R, DONAHUE J, DARRELL T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2014: 580-587.

[16] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[J]. arXiv Preprint, arXiv: 1409.1556, 2014.

[17] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[C]//Advances in Neural Information Processing Systems. Piscataway: IEEE Press, 2015: 91-99.

[18] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: unified, real-time object detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2016: 779-788.

[19] XIONG J B, ZHAO M, BHUIYAN M, et al. An AI-enabled three-party game framework for guaranteed data privacy in mobile edge crowdsensing of IoT[J]. IEEE Transactions on Industrial Informatics, 2019, doi: 10.1109/TII.2019.2957130.

[20] GENTRY C. Fully homomorphic encryption using ideal lattices[C]//Proceedings of the 41th Annual ACM Symposium on Theory of Computing. New York: ACM Press, 2009: 169-178.

[21] DOWLIN N, GILAD-BACHRACH R, LAINE K, et al. Cryptonets: applying neural networks to encrypted data with high throughput and accuracy[C]//International Conference on Machine Learning. New York: ACM Press, 2016: 201-210.

[22] HESAMIFARD E, TAKABI H, GHASEMI M. CryptoDL: deep neural networks over encrypted data[J]. arXiv Preprint, arXiv:1711.05189, 2017.

[23] JUVEKAR C, VAIKUNTANATHAN V, CHANDRAKASAN A. Gazelle: a low latency framework for secure neural network inference[C]//27th USENIX Security Symposium. Berkeley: USENIX Association, 2018: 1651-1669.

[24] HUANG K, LIU X M, FU S J, et al. A lightweight privacy-preserving CNN feature extraction framework for mobile sensing[J]. IEEE Transactions on Dependable and Secure Computing, 2019, doi: 10.1109/TDSC.2019.2913362.

[25] LIU Y, MA Z, LIU X M, et al. Privacy-preserving object detection for medical images with Faster R-CNN[J]. IEEE Transactions on Information Forensics and Security, 2019(10): 1.

[26] LI F H, LI H, NIU B, et al. Privacy computing: concept, computing framework, and future development trends[J]. Engineering, 2019, 5(6): 981-1192.

[27] BELANOVIĆ P, LEESER M. A library of parameterized floating-point modules and their use[C]//International Conference on Field Programmable Logic and Applications. Berlin: Springer, 2002: 657-666.

[28] XIONG J B, BI R, ZHAO M, et al. Edge-assisted privacy-preserving raw data sharing framework for connected autonomous vehicles[J]. IEEE Wireless Communications, 2020, 27(3): 24-30.

[29] BOGDANOV D, LAUR S, WILLEMSON J. Sharemind: a framework for fast privacy-preserving computations[C]//European Symposium on Research in Computer Security. Berlin: Springer, 2008: 192-206.

[30] BOGDANOV D, NIITSOO M, TOFTETAL T, et al. High-performance secure multi-party computation for data mining applications[J]. International Journal of Information Security, 2012, 11(6): 403-418.

[31] MA Z, LIU Y, LIU X, et al. Privacy-preserving outsourced speech recognition for smart IoT devices[J]. IEEE Internet of Things Journal, 2019, 6(5): 8406-8420.

[32] EVERINGHAM M, VANGOOL L, WILLIAMS C, et al. The PASCAL visual object classes challenge 2007 (VOC2007) results[C]//International Conference on Computer Vision. Piscataway: IEEE Press, 2007: 1-24.

[作者简介]

熊金波（1981– ），男，湖南益阳人，博士，福建师范大学教授，主要研究方向为安全深度学习、移动群智感知、隐私保护技术等。

毕仁万（1996– ），男，湖南常德人，福建师范大学硕士生，主要研究方向为安全深度学习、安全多方计算等。

陈前昕（1996– ），男，福建泉州人，福建师范大学硕士生，主要研究方向为安全深度学习、隐私保护技术等。

刘西蒙（1988– ），男，陕西西安人，博士，福州大学教授，主要研究方向为云安全、应用密码学、大数据安全等。